VYSOKÉ UČENÍ TECHNICKÉ V BRNĚ
BRNO UNIVERSITY OF TECHNOLOGY

FAKULTA INFORMAČNÍCH TECHNOLOGIÍ
ÚSTAV INFORMAČNÍCH SYSTÉMŮ

FACULTY OF INFORMATION TECHNOLOGY
DEPARTMENT OF INFORMATION SYSTEMS

MONITOROVÁNÍ A ÚČTOVÁNÍ SPOJENÍ V SÍTÍCH IMS

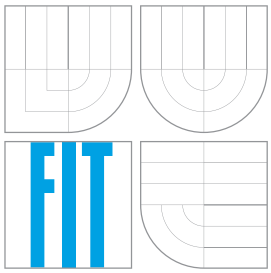
DIPLOMOVÁ PRÁCE
MASTER'S THESIS

AUTOR PRÁCE
AUTHOR

Bc. FILIP KARPÍŠEK

BRNO 2015

VYSOKÉ UČENÍ TECHNICKÉ V BRNĚ
BRNO UNIVERSITY OF TECHNOLOGY



FAKULTA INFORMAČNÍCH TECHNOLOGIÍ
ÚSTAV INFORMAČNÍCH SYSTÉMŮ

FACULTY OF INFORMATION TECHNOLOGY
DEPARTMENT OF INFORMATION SYSTEMS

MONITOROVÁNÍ A ÚČTOVÁNÍ SPOJENÍ V SÍTÍCH IMS

SESSION MONITORING AND ACCOUNTING IN IMS NETWORKS

DIPLOMOVÁ PRÁCE
MASTER'S THESIS

AUTOR PRÁCE
AUTHOR

Bc. FILIP KARPÍŠEK

VEDOUCÍ PRÁCE
SUPERVISOR

Ing. PETR MATOUŠEK, Ph.D.

BRNO 2015

Abstrakt

Tato práce popisuje protokoly používané v sítích IP Multimedia Subsystem (IMS). Zabývá se také volně dostupnými implementacemi tohoto systému. Hlavním cílem této práce je popsat návrh a implementaci nástroje pro analýzu komunikace mezi systémem a uživateli. Nástroj vyhledává a dekóduje signalizační zprávy, ze kterých získá informace o spojení, které jsou nutné pro monitorování a účtování těchto spojení. Zpracované informace jsou exportovány ve formě rozšířených záznamů NetFlow/IPFIX. Pro vytvoření sítě IMS a testovacích dat jsme využili volně dostupné ústředny Open IMS Core. Jako koncové body jsme použili rovněž volně dostupnou aplikaci IMSDroid pro operační systém Android.

Abstract

This thesis describes protocols used in IP Multimedia Subsystem (IMS) networks. Freely available implementations of IMS system are described. The main goal is to describe design and implementation of a tool for analyzing communication between users and IMS system. The tool seeks and decodes signaling messages. These messages are analyzed for information about sessions which are necessary for session monitoring and accounting. Final gathered information are exported in a form of extended NetFlow/IPFIX records. We used open-source Open IMS Core implementation for building IMS network and creating test data. As endpoints we used another open-source application for Android OS called IMSDroid.

Klíčová slova

LTE, IMS, SIP, IPFIX, NetFlow, Open IMS Core, IMSDroid

Keywords

LTE, IMS, SIP, IPFIX, NetFlow, Open IMS Core, IMSDroid

Citace

Filip Karpíšek: Monitorování a účtování spojení v sítích IMS, diplomová práce, Brno, FIT VUT v Brně, 2015

Monitorování a účtování spojení v sítích IMS

Prohlášení

Prohlašuji, že jsem tento semestrální projekt vypracoval samostatně pod vedením pana Ing. Petra Matouška, Ph. D.
Uvedl jsem všechny literární prameny a publikace, ze kterých jsem čerpal.

.....
Filip Karpíšek
21. července 2015

Poděkování

Tímto chci poděkovat Ing. Petrovi Matouškovi, Ph. D. jako vedoucímu mé diplomové práce za jeho rady a odbornou pomoc, kterou mi při vedení této práce poskytl. Dále bych chtěl poděkovat Martinu Elichovi ze společnosti INVEA-TECH za pomoc s implementací pluginu pro FlowMon sondu.

Obsah

Seznam použitých zkratek

AuC	Authentication Center
CDMA2000	Code Division Multiple Access 2000
CSCF	Call Session Control Function
DNS	Domain Name System
DSL	Digital Subscriber Line
GPRS	General Packet Radio Service
GRUU	Globally Routable User Agent URI
GSM	Global System for Mobile
HLR	Home location register
HSS	Home Subscriber Server
I-CSCF	Interrogating-CSCF
IMPI	IP Multimedia Private Identity
IMPU	IP Multimedia Public Identity
IMS	IP Multimedia Subsystem
IPFIX	Internet Protocol Flow Information Export
IP	Internet Protocol
LTE	Long-Term Evolution
MTU	Maximum Transmission Unit
P-CSCF	Proxy-CSCF
P-GRUU	Permament GRUU
PDA	Personal Digital Assistant
RTP	Real-time Transport Protocol
S-CSCF	Serving-CSCF
SCTP	Stream Control Transmission Protocol
SIP	Session Initiation Protocol
T-GRUU	Temporary GRUU
TCP	Transmission Control Protocol
UDP	User Datagram Protocol
UE	User Equipment
URI	Uniform Resource Identifier
VoIP	Voice over Internet Protocol
WiMAX	Worldwide Interoperability for Microwave Access
WLAN	Wireless Local Area Network

Kapitola 1

Úvod

Mobilní telefon a podobná mobilní zařízení už dávno nevyužívají pouze hlasové služby mobilní sítě. Čím dál větší počet zařízení je využíván k přístupu ke službám sídlícím na Internetu mimo infrastrukturu poskytovatele. To vede k postupnému snižování využití prvků poskytujících služby v síti poskytovatele a tato síť se stává pouze sítí tranzitní.

Systém IMS [1] se snaží přesunout tyto služby zpět do sítě poskytovatele, neboť se jedná především o služby multimediální, ať už jsou to hovory VoIP, video-hovory či sdílení dat.

Tato práce vznikla za podpory grantu MV *Moderní prostředky pro boj s kybernetickou kriminalitou na Internetu nové generace* (VG20102015022).

1.1 Motivace

Systém IMS dokáže poskytovat celou řadu služeb. Tyto služby lze analyzovat a na základě takovéto analýzy následně provádět monitorování případě účtování těchto služeb. Vzhledem k tomu, že veškerá komunikace probíhá nad protokolem IP, je možné provádět analýzu této komunikace na vnějším rozhraní systému IMS, což může být vhodné zejména tam, kde nemůžeme do tohoto systému zasáhnout a přesto bychom rádi monitorování prováděli. Samotné monitorování je přínosné např. pro přehled komunikace, detekci útoků, sledování využití služeb a v neposlední řadě k účtování. Účtování spojení je pak možné provádět na základě informací získaných z analýzy spojení.

V současné době je systém IMS převážně nasazován u velkých zahraničních operátorů[1]. V Česku jej využívají například operátoři Telefonica O2[2] a T-Mobile[3].

[1] http://www.thefastmode.com/ip-multimedia-subsystem-deployments

[2] http://www.feedit.cz/wordpress/2012/09/24/ericsson-a-telefonica-ceska-republika-podepsali-smlouvu-na-dodavku-rozsireni-systemu-ims-ip-multimedia-subsystem-pro-ip-komunikaci/

[3] http://www.thefastmode.com/technology-solutions/3293-t-mobile-czech-republic-deploys-mavenir-systems-ims-core-volte-vas-system

1.2 Cíl

Cílem této práce je podrobně prozkoumat komunikaci systému IMS se svým okolím a možností analýzy této komunikace, dále navrhnout, implementovat a otestovat nástroj, který by prováděl monitorování a účtování.

Přínos takového nástroje pak spočívá v automatické transformaci síťového provozu na množinu uživatelsky užitečných informací, čímž se rozumí soubor informací o jednotlivých spojeních – jejich začátek a konec, množství přenesených dat vytvořeným kanálem, atd. Takové informace jsou přínosné při správě sítě, přičemž je vhodné zaintegrovat takový nástroj do standardních monitorovacích nástrojů, například NetFlow/IPFIX [3], aby nebylo potřeba množinu těchto monitorovacích nástrojů dále rozšiřovat.

1.3 Návrh a postup řešení

K vytváření spojení IMS je typicky využíván protokol SIP [7]. Prvním krokem je tedy zjištění relevantních informací přenášených v protokolu SIP a dekódování zpráv protokolu SIP k získání informací o spojení.

Tyto informace bude nutné dále propojit. Například pro informaci o délce spojení je nutné detekovat začátek a konec spojení, pro informace o množství přenesených dat je nutné identifikovat datové toky apod.

Práci na vývoji tedy rozdělíme do tří částí:

1. návrh prostředí systému IMS, vytvoření struktury systému IMS a jeho konfigurace
2. analýza přenosů a spojení IMS, detekce těchto spojení, extrakce dat o spojeních IMS
3. zpracování a uložení dat o spojeních IMS

Tato práce je rozvržena následujícím způsobem. V kapitole 2 rozebíráme systém IMS jako takový. V kapitole 3 pak uvádíme volně dostupné implementace systému IMS, které jsme použili pro vytvoření sítě IMS a rovněž nutnou konfiguraci systému IMS. Návrh a implementaci nástroje pro monitorování a účtování spojení v sítích IMS popisujeme v kapitole 4 a v kapitole 5 pak způsob a výsledky testování tohoto nástroje.

Kapitola 2

Systém IMS (IP Multimedia Subsystem)

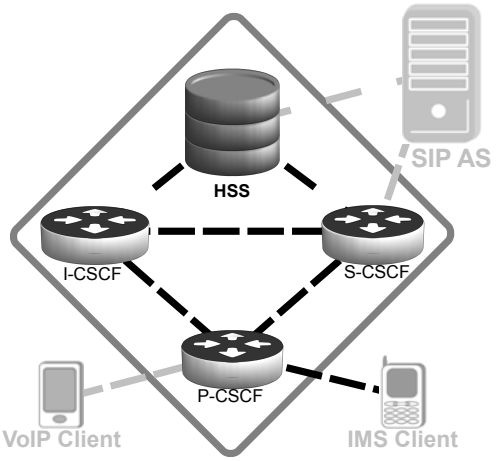
Systém IMS je množinou rozličných funkcí, které jsou propojeny standardizovanými rozhraními a slouží k doručování multimediálních služeb přes síť IP. Tyto funkce dohromady tvoří administrativní síť IMS. Jedna funkce není prováděna uzlem (z hardwarového hlediska) – je možné funkce kombinovat či naopak štěpit do více uzlů, případně funkce duplikovat (ať už z důvodů škálovatelnosti, vyvažování zátěže, organizačních nebo jiných).

Uživatelé se mohou do sítě systému IMS připojit různými způsoby, přičemž většina z nich využije běžné sítě IP. Připojovaná zařízení zvaná terminály (mobilní telefony, zařízení PDA, počítače, ...) se mohou registrovat přímo do sítě IMS, i když jsou z jiné sítě či země. Jediným požadavkem je, že dokáží používat síť IP a může na nich fungovat klient SIP. Jsou podporovány technologie pro pevné připojení (DSL, kabelový modem, Ethernet, atd.), mobilní připojení (W-CDMA, CDMA2000, GSM, GPRS, atd.) i bezdrátové připojení (WLAN, WiMAX, atd.). Další telefonní systémy jako veřejná telefonní síť, H.323 a další systémy nekompatibilní s IMS mohou být připojeny pomocí speciálních bran.

Obecná architektura páteřní části systému IMS je popsána na obrázku 2.1. Jednotlivé funkce budou popsány dále. Tato síť obsahuje koncové body: VoIP client a IMS client; přenosovou (administrativní) síť: HSS a CSCF a uzel poskytující služby: SIP AS.

2.1 Adresář HSS (Home Subscriber Server)

Jedná se o hlavní databázi uživatelů, která slouží funkcím, které se starají u samotné hovory a jiná spojení. Obsahuje informace, které se vztahují k množině dostupných služeb (subscriber profiles). HSS provádí autentizaci a autorizaci uživatelů, může poskytovat informace o jejich poloze, parametrech IP a další. Celkově se HSS dá připodobnit k funkcím Home



Obrázek 2.1: Obecná architektura systému IMS

Location Register (HLR) a Authentication Center (AuC) v sítích GSM.

2.1.1 Uživatelské identity

V síti IMS se mohou vyskytovat různé identity. Jedná se o veřejnou a soukromou identitu, globálně směrovatelnou URI a veřejnou identitu s maskou. Tyto identity jsou důležité pro správu a účtování. První dvě zmíněné identity nejsou telefonní čísla ani sekvence číslic, jak je tomu v GSM síti, ale libovolné řetězce fungující jako uživatelská jména. Identifikátor URI však může být jak telefonním číslem (např. `tel:+420-602-000-001`) tak sekvencí alfanumerických znaků (např. `sip:jan.novak@domena.cz`).

Soukromá identita (IP Multimedia Private Identity – IMPI)

IMPI je jedinečná permanentní globální identita, kterou přiřazuje operátor domácí sítě uživatelům po celou dobu, kdy se uživatel vyskytuje v síti operátora. Je určena k registraci, autorizaci a administrativnímu přístupu uživatele a je vhodná k monitorování a účtování, protože každý uživatel musí mít právě jednu takovouto identitu.
Příklad: `sip:jan.novak.soukrome@domena.cz`

Veřejná identita (IP Multimedia Public Identity – IMPU)

Tato identita je použita při komunikaci mezi uživateli. Jedna soukromá identita (IMPI) může mít přiřazeno více veřejných identit (IMPU). Jedna veřejná identita může být sdílena mezi více zařízení, takže více zařízení může být adresováno stejnou veřejnou identitou (např. jedno telefonní číslo pro celou rodinu).

Příklad: `sip:jan.novak@domena.cz`, `sip:obchod@firma.cz`

Globálně směrovatelná URI uživatele (GRUU)

Identifikuje unikátní kombinaci veřejné identity (IMPU) a klientského zařízení – User Equipment (UE). Při použití GRUU nedochází k duplikaci požadavků protokolu SIP na různá zařízení registrovaná pod stejnou veřejnou identitou.
Existují dva typy GRUU:

- P-GRUU: veřejná, s dlouhou životností, obsahuje veřejnou identitu
- T-GRUU: dočasné, neobsahuje veřejnou identitu a má platnost pouze po dobu registrace

Příklad P-GRUU: `sip:jan.novak@domena.cz;gr=kjh29x97us97d`
Příklad T-GRUU: `sip:asd887f9dfkk76690@domena.cz;gr`

Veřejná identita s maskou (Wildcarded Public User Identity)

Veřejná identita s maskou vyjadřuje množinu veřejných identity seskupených dohromady.
Příklad: `sip:*@obchod.firma.cz`

2.2 Funkce CSCF (Call Session Control Function)

Tato funkce může nabývat různých rolí v rámci systému IMS. V následujících oddílech budou tyto role popsány. Tyto role potřebujeme znát, abychom mohli pokrýt všechny možné scénáře komunikace a podle toho navrhnout nástroj pro monitorování.

Role P-CSCF (Proxy-CSCF)

Proxy-CSCF je prvním kontaktním bodem systému IMS s UE. Získání její adresy může být provedeno následujícími způsoby:

- předáním v rámci procesu připojování UE do sítě a konfigurace tohoto připojení (např. pomocí DHCP – option 120)
- stejně, jako v předchozím případě, jen s tím rozdílem, že místo adresy je předáno doménové jméno, které je později pomocí DNS přeloženo na adresu
- UE je pevně nakonfigurováno, zná tedy doménové jméno nebo adresu

Role P-CSCF se chová jako Proxy (definováno v RFC 3261 [7]), takže přijímá požadavky, které zpracovává, případně směruje dále do systému. Pro naše potřeby monitorování a účtování je dostatečné analyzovat komunikaci mezi P-CSCF a UE.

Role I-CSCF (Interrogating-CSCF) a S-CSCF (Serving-CSCF)

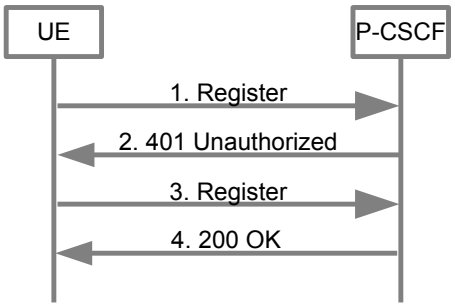
I-CSCF a S-CSCF zajišťují spojení mezi uživateli uvnitř i vně sítě (i od různých operátorů) a tato spojení také řídí. Pro účely monitorování a účtování spojení však stačí monitorovat komunikaci mezi UE a P-CSCF.

2.3 Komunikace UE a P-CSCF

Pro monitorování a následné účtování spojení je nutné monitorovat zprávy na hranici sítě IMS směrem k uživateli, tj. mezi UE a funkcí P-CSCF. Následuje přehled scénářů, které mohou mezi těmito dvěma entitami nastat.

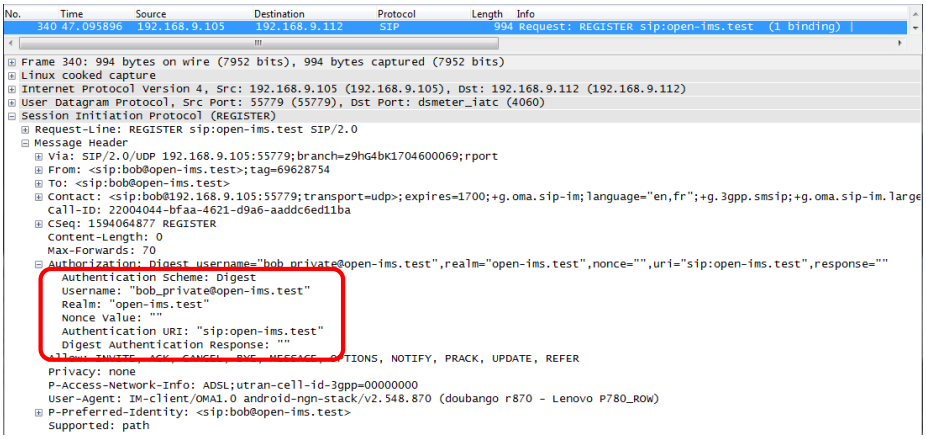
2.3.1 Přihlášení uživatele (dosud nepřihlášený)

Na obrázku 2.2 jsou zobrazeny zprávy, které jsou vyměněny mezi UE a funkcí P-CSCF během registrace dosud nezaregistrovaného uživatele.

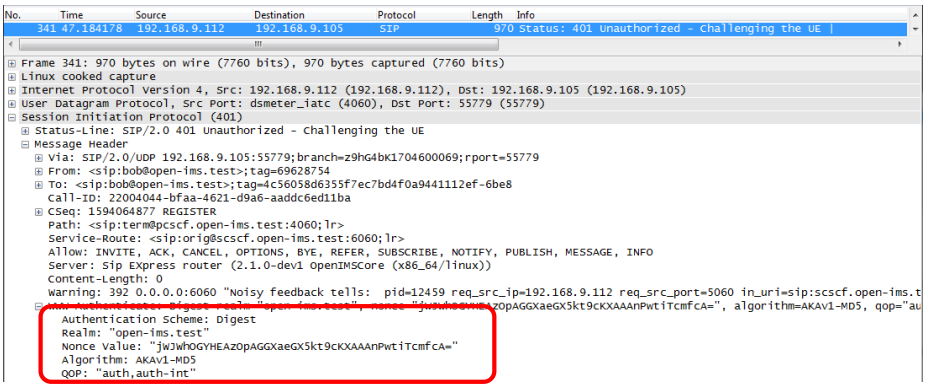


Obrázek 2.2: Komunikace UE a P-CSCF: registrace nezaregistrovaného uživatele

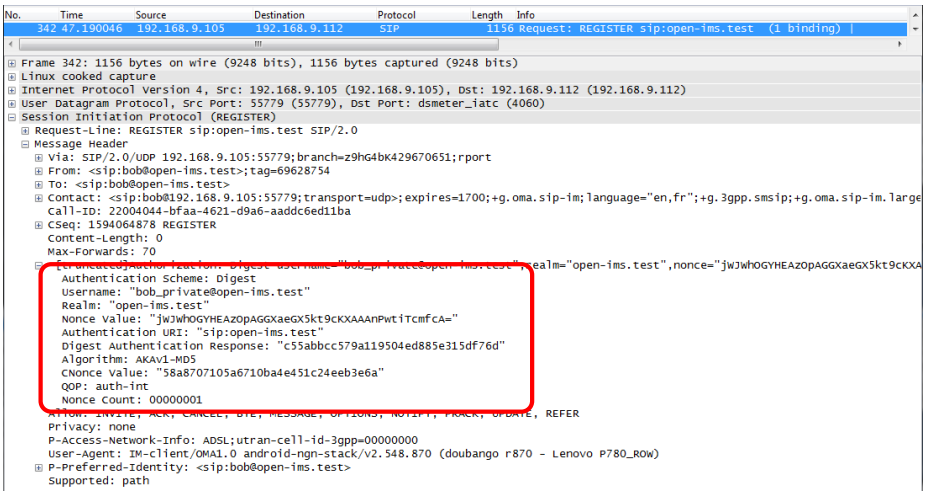
UE se nejprve pokusí registrovat pouze s uživatelským jménem. Ukázka takové zprávy je na obrázku 2.3. Uvnitř položky `Authorization` můžeme v parametru `Username` najít soukromou identitu (IMPI) – `bob_private@open-ims.test`. Poté je zprávou `401 Unauthorized` UE vyzván, aby odpověděl na autentizační výzvu. Ukázka zprávy `401 Unauthorized` je na obrázku 2.4. UE tedy odpoví, viz druhá zpráva `Register` na obrázku 2.5, a dostává zpět zprávu `200 OK`, viz obrázek 2.6.



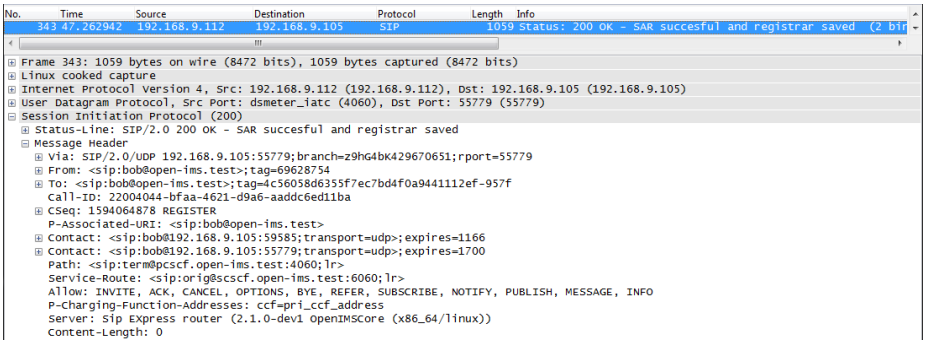
Obrázek 2.3: Zpráva 1. Register



Obrázek 2.4: Zpráva 2. 401 Unauthorized



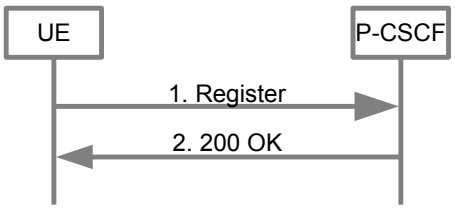
Obrázek 2.5: Zpráva 3. Register



Obrázek 2.6: Zpráva 4. 200 OK

2.3.2 Přihlášení uživatele (již přihlášený)

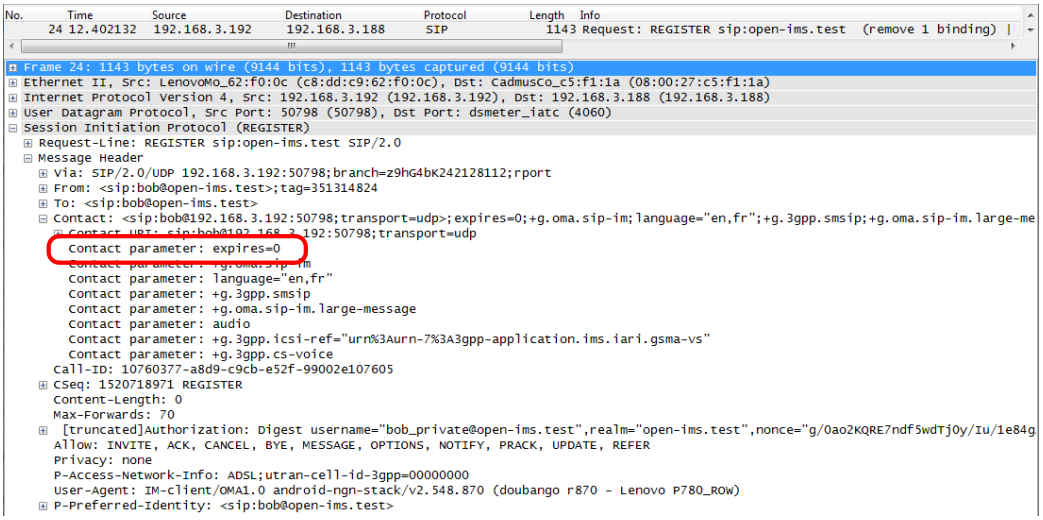
Na obrázku 2.7 jsou zobrazeny zprávy, které jsou vyměněny mezi UE a funkcí P-CSCF během registrace již zaregistrovaného uživatele. Tato komunikace probíhá podobně jako druhá polovina předchozí komunikace.



Obrázek 2.7: Komunikace UE a P-CSCF: registrace již zaregistrovaného uživatele

2.3.3 Odhlášení uživatele

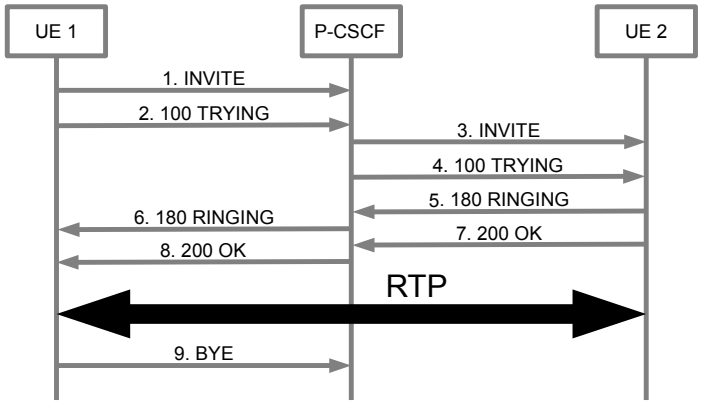
Odhlášení uživatele probíhá naprosto stejně, jako přihlášení dosud neregistrovaného uživatele. Rozdíl je pouze v tom, že parametr `expires` v položce `Contact` je nastaven na hodnotu 0, viz zobrazená zpráva `Register` na obrázku 2.8.



Obrázek 2.8: Zpráva Register při odhlašování uživatele

2.3.4 Vytvoření a průběh spojení

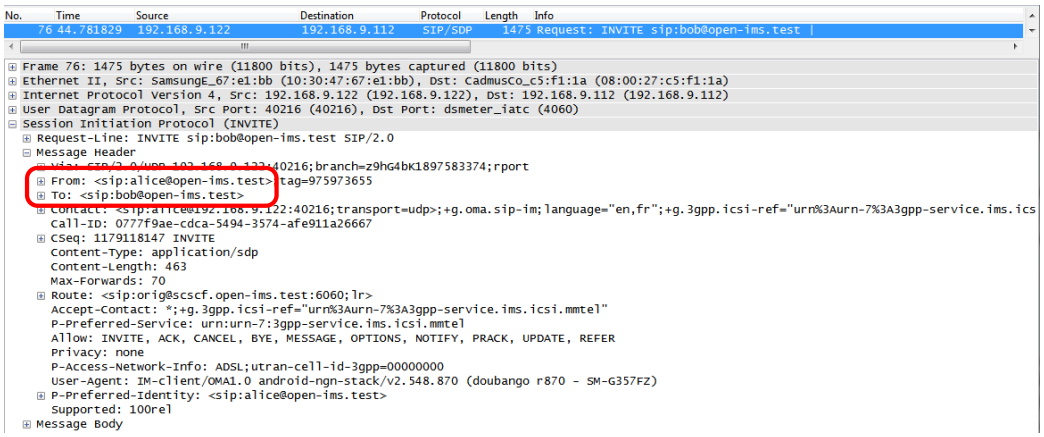
Na obrázku 2.9 jsou zobrazeny zprávy a tok RTP [8] mezi uživatelskými zařízeními a funkcí P-CSCF při vytváření a průběhu spojení mezi těmito UE. Pro adresaci (položky `From` a `To` ve zprávách protokolu SIP) jsou použity veřejné identity (IMPU), například `sip:bob@open-ims.test` a `sip:alice@open-ims.test`.



Obrázek 2.9: Komunikace UE a P-CSCF: vytvoření a průběh spojení

UE 1 zasílá zprávu `INVITE` protokolu SIP funkci P-CSCF, která vyhledá cíl této zprávy podle veřejné identity adresáta v položce `To` v hlavičce této zprávy. Systém IMS si uchovává informace o všech přihlášených zařízeních (včetně jejich IP adres a portů), podle kterých může signalizační zprávy protokolu SIP směrovat. Zprávy `INVITE` a `200 OK` obsahují v těle zprávy protokol SDP [4], ve kterém je popsán tok RTP, který přenáší samotná multimediální

data. Ukázka takové zprávy s vyznačenými identitami je na obrázku 2.10.



Obrázek 2.10: Zpráva INVITE při navazování spojení

2.4 Shrnutí

V této kapitole jsme si popsali fungování systému IMS a detaily komunikace mezi systémem IMS a jeho okolím, které jsou podstatné pro detekci a monitorování této komunikace. Zjistili jsme, že pro detekci uživatelů jsou pro nás podstatné zejména zprávy `REGISTER` a následná zpráva `200 OK` a pro detekci samotných spojení pak zprávy `INVITE` a příslušná zpráva `200 OK`. Dále v kapitole 3 popíšeme vytvoření samotného systému IMS, který využijeme k testování našeho nástroje.

Kapitola 3

Vytvoření sítě IMS

Tato kapitola popisuje instalaci, konfiguraci a použití volně dostupné implementace systému IMS, ústřednu Open IMS Core[1]. Tato ústředna byla použita k vytvoření spojení IMS a jejich zachycení za účelem analýzy spojení IMS. Zachycená spojení byla použita k testování vytvořeného nástroje.

3.1 Ústředna Open IMS Core

Jedná se o volně dostupnou implementaci systému IMS. Tuto ústřednu lze nainstalovat na běžný unixový operační systém a používat například v kombinaci s klientem IMS nainstalovaném na mobilním zařízení, viz kapitola 3.2.

Na obrázku 3.1 je zobrazena základní architektura této ústředny. Po instalaci nevyžaduje již žádné další komponenty. Jako aplikační server SIP využívá SIP Express Router[2].

Instalace bylo provedena podle instalačního návodu na webu Open IMS Core[3], byl použit operační systém Kubuntu ve verzi 14.10. Po dokončení instalace bylo nutné upravit konfigurační soubory jednotlivých funkcí (P-CSCF, I-CSCF a S-CSCF), protože ve výchozím nastavení očekávají spojení na adrese loopback (127.0.0.1). To je nevhodné, pokud chceme ústřednu používat pro spojení mezi zařízeními. Za tímto účelem je třeba upravit konfigurační skripty výše zmíněných funkcí a konfiguraci domény DNS. Upravili jsme tyto soubory:

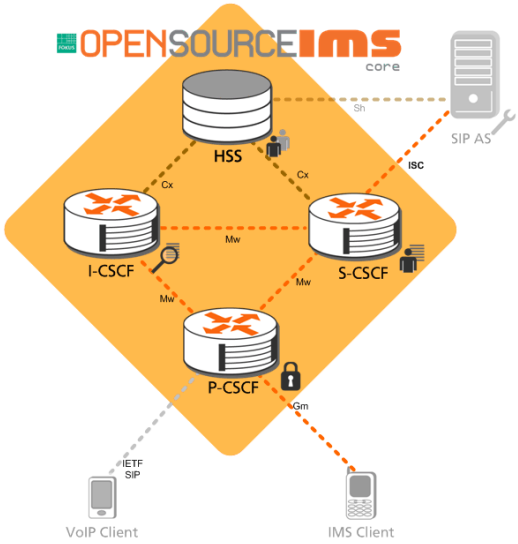
`/opt/OpenIMSCore/pcscf.cfg`

`/opt/OpenIMSCore/icscf.cfg`

`/opt/OpenIMSCore/scscf.cfg`

`/etc/bind/open-ims.dnszone`

[1] http://www.openimscore.org/
[2] http://www.iptel.org/ser/
[3] http://www.openimscore.org/?q=installation_guide



Obrázek 3.1: Architektura ústředny Open IMS Core, převzato z webu Open IMS Core [1]

Konfigurační soubory jednotlivých funkcí stačí upravit tak, že řádek, který obsahuje parametr `listen=127.0.0.1` změníme tak, aby obsahoval adresu vnějšího rozhraní stroje, na kterém jsme provedli instalaci. Obdobným způsobem upravíme konfigurační soubor zóny DNS tak, aby řádky začínající `pcscf`, `icscf` a `scscf` obsahovaly adresu vnějšího rozhraní namísto adresy `127.0.0.1`, viz následující ukázka následující ukázka konfigurace DNS serveru `bind` (pouze část souboru `/etc/bind/open-ims.dnszone`, adresa vnějšího rozhraní je v tomto případě `192.168.9.112`):

```
...
pcscf                   1D IN A         192.168.9.112
_sip.pcscf              1D SRV 0 0 4060 pcscf
_sip._udp.pcscf         1D SRV 0 0 4060 pcscf
_sip._tcp.pcscf         1D SRV 0 0 4060 pcscf

icscf                   1D IN A         192.168.9.112
_sip                    1D SRV 0 0 5060 icscf
_sip._udp               1D SRV 0 0 5060 icscf
_sip._tcp               1D SRV 0 0 5060 icscf

scscf                   1D IN A         192.168.9.112
_sip.scscf              1D SRV 0 0 6060 scscf
_sip._udp.scscf         1D SRV 0 0 6060 scscf
_sip._tcp.scscf         1D SRV 0 0 6060 scscf
...
```

Záznamy `SRV` určují záznamy DNS pro služby poskytované síťovými uzly. Formát těchto záznamů je následující:

```
_service._prot.name. TTL class SRV priority weight port target.
```

Položky v záznamu `SRV` a jejich význam:

- `service` symbolický název služby
- `prot` transportní protokol služby (většinou TCP nebo UDP)
- `name` doménové jméno, pro kterou je tento záznam platný, ukončené tečkou
- `TTL` délka platnostni záznamu (standardní položka DNS)
- `class` třída záznamu, je vždy `IN` (standardní položka DNS)
- `priority` priorita záznamu, méně znamená prioritnější
- `weight` relativní váha pro záznamy se stejnou prioritou, více znamená prioritnější
- `port` číslo portu služby, na kterém se služba vyskytuje
- `target` kanonické doménové jméno uzlu poskytujícího službu, ukončené tečkou

Výsledkem dotazu na tento typ záznamu je číslo portu a adresa uzlu, na kterém se daná služba vyskytuje. Na záznamy typu `SRV` je možné se dotazovat běžnými nástroji pro síťovou administrativu (například `nslookup`). Dotaz například na službu `_sip` uzlu `pcscf` a jeho výsledek vypadá následovně:

```
$ nslookup -querytype=srv _sip.pcscf.open-ims.test
Server:         127.0.1.1
Address:        127.0.1.1#53

_sip.pcscf.open-ims.test
        service = 0 0 4060 pcscf.open-ims.test.
```

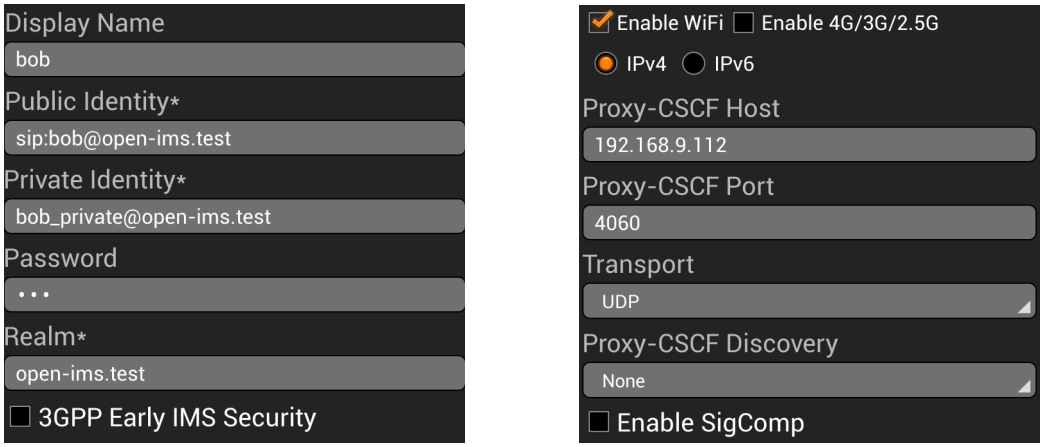
3.2 Klient IMS – IMSDroid

Aplikace IMSDroid[4] pro operační systém Android umožňuje propojení s ústřednou Open IMS Core a využívání jejích služeb. Tuto aplikaci jsme zvolili, protože je volně dostupná a poskytuje veškerou funkcionalitu, kterou potřebujeme.

Pro provedení všech scénářů potřebujeme mít do systému zapojena dvě klientská zařízení. První zařízení nazýváme UE Alice, přihlašuje se pomocí IMPI `alice_private@open-ims.test` a IMPU `alice@open-ims.test`. Druhé zařízení nazýváme UE Bob a přihlašuje se obdobnými identifikátory: IMPI `bob_private@open-ims.test` a IMPU `bob@open-ims.test`.

Konfigurace aplikace IMSDroid pro UE Bob je zobrazena na obrázku 3.2, vlevo je konfigurace identity uživatele, vpravo konfigurace sítě – adresa `192.168.9.112` odpovídá vnějšímu rozhraní stroje s ústřednou Open IMS Core.

[4] https://code.google.com/p/imsdroid/



Obrázek 3.2: Konfigurace aplikace IMSDroid

3.2.1 Podporované služby

Klient IMSDroid podporuje následující služby systému IMS:

- běžný (audio) a videohovor
- instant messaging – chat
- sdílení souborů

3.3 Shrnutí

V této kapitole jsme si popsali vytvoření a konfiguraci systému IMS a klientských zařízení, která jsme v rámci tohoto systému používali k provedení scénářů. Při provádění těchto scénářů jsme zachytávali síťovou komunikaci, která bude použita pro testování nástroje. Vytváření testovacích dat a provádění testů popíšeme v kapitole 5. V následující kapitole 4 popíšeme návrh a implementaci nástroje pro monitorování a účtování spojení v sítích IMS.

Kapitola 4

Nástroj pro monitorování a účtování

V této kapitole popíšeme návrh a implementaci nástroje pro monitorování a účtování spojení IMS. Nástroj je možné implementovat dvěma způsoby:

1. rozšířením funkce ústředny Open IMS Core
2. nezávislým nástrojem monitorující rozhraní libovolného systému IMS bez nutnosti zásahu do systému IMS, například řešení FlowMon [5], které umožňuje monitorování sítí na bázi toků NetFlow/IPFIX

Nakonec jsme zvolili druhou variantu z následujících důvodů:

- ústředna Open IMS Core slouží spíše k výuce a experimentům
- druhá varianta je univerzálnější, pokrývá více než jen ústřednu Open IMS Core
- není nutné nijak zasahovat do systému IMS, což může být vhodné v situacích, kdy to není možné (systém je již nasazený, je uzavřený, nemáme oprávnění, apod.)

4.1 Návrh nástroje – plugin pro sondu FlowMon

Nástroj je implementován v prostředí sondy FlowMon, takže před vlastním návrhem bylo nutné se seznámit s technologií a architekturou sondy a možností vývoje software pro tuto platformu.

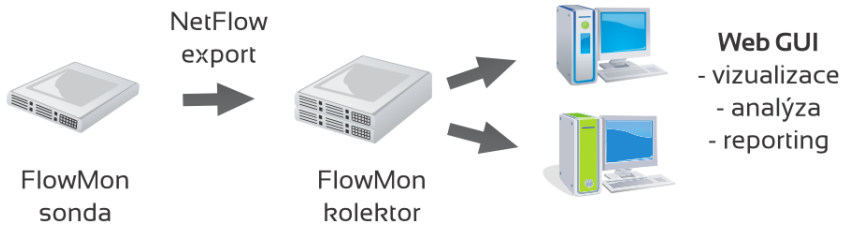
Pro úspěšné monitorování a účtování spojení je nutné vykonat tyto úkoly:

1. detekce a zpracování přihlašování a odhlašování uživatelů (vazba IMPI na IMPU)
2. detekce a zpracování samotných spojení včetně datových toků (adresace pomocí IMPU či GRUU)

3. provázání IMPU nebo GRUU použitých v kroku 2 s IMPI získanými v kroku 1
4. export dat

4.1.1 Sonda FlowMon

Sonda FlowMon [6] patří do portfolia produktů FlowMon [5] společnosti INVEA-TECH tvořící kompletní řešení pro monitorování sítí na základě toků. Řešení se sestává ze sond, kolektorů a rozšiřujících pluginů pro sondy a kolektory. Sondy FlowMon analyzují každý procházející paket a na základě těchto dat generují NetFlow/IPFIX statistiky, které jsou exportovány na kolektor pro další zpracování a vizualizaci. Zjednodušená architektura celého řešení je zobrazena na obrázku 4.1.



Obrázek 4.1: Zjednodušená architektura řešení FlowMon [6]

Formát exportovaných NetFlow/IPFIX záznamů je možné rozšiřovat o libovolné hodnoty a tím přidávat vlastní proprietární informace do záznamů. Vývoj pluginů pro sondy FlowMon probíhá v rámci komunitního programu na virtuální sondě FlowMon. Je možné vytvářet pluginy různých typů podle způsobu práce s toky:

- vstupní plugin — analýza paketů a získávání potřebných dat
- procesní plugin — zpracování informací ze vstupní části a aktualizace záznamů flow cache
- exportní plugin — filtrace záznamů a export na kolektor

Každý plugin si při své inicializaci vytvoří svou privátní strukturu, která je přístupná po celou dobu běhu pluginu. Rovněž je umožněno přidat ke každému toku privátní strukturu, která umožňuje uchovávat proprietární data toku, se kterými pak plugin pracuje. Položky rozšířených záznamů toků jsou pak naplňovány takzvanými „`gettery`“. Getter se skládá z těcho čtyř funkcí:

- `valid` – funkce určující, zda je položka platná
- `length` – funkce udávající aktuální délku položky v bytech
- `filler` – funkce naplňující položku hodnotou
- `filler_txt` – funkce naplňující textovou reprezentaci hodnoty položky

4.1.2 Fáze 1 – detekce a analýza registračních událostí

Tato fáze slouží k získání vazby mezi IMPI a IMPU. IMPU se používá k adresaci jednotlivých spojení a IMPI k identifikaci uživatele. Právě IMPI vyžadujeme pro monitorování a účtování spojení, takže je nutné tuto vazbu získat.

V síťovém toku budeme vyhledávat signalizační zprávy `REGISTER` protokolu SIP, a to ty, které jsou směrované na nebo z portu 4060, na kterém komunikuje funkce P-CSCF. Z těchto zpráv získáme z položek `From` a `To` veřejnou identitu IMPU a z parametru `username` položky `Authorization` pak soukromou identitu IMPI. Tyto parametry odpovídají hodnotám v tabulce 4.2. Budeme rovněž sledovat, zda byla registrace úspěšná. Neúspěšné registrace vazbu mezi IMPI a IMPU neposkytují.

Signalizační zprávy `REGISTER` obsahují ještě další zajímavé položky. Tyto položky mohou poskytovat další informace, které mohou být pro správce sítě užitečné, například typ přístupové sítě (položka `P-Access-Network-Info`), klientská aplikace (položka `User-Agent`) a další. Tyto položky nebudeme v současné verzi nástroje nijak využívat, mohly by však být využity při jeho rozšiřování.

4.1.3 Fáze 2 – detekce a analýza spojení

Podobně jako v předchozí fázi i v této fázi budeme sledovat signalizační zprávy protokolu SIP, tentokrát však zprávy typu `INVITE` a následující `200 OK`, neboť tyto zprávy obsahují i informace o tocích RTP. Tyto toky budeme rovněž sledovat pro případ, že by bylo požadováno účtování na základě přenesených dat, případě monitorování kvality přenosu dat (rozptyl, ztrátovost, atp.). Na základě informací získaných ze signalizace to bude možné. Položky záznamů těchto toků jsou uvedeny v tabulce 4.3. V této fázi budeme naplňovat položku `address`, jednu z položek `port`, `videoPort` nebo `msrpPort` (podle typu toku – zbylé z těcho tří položek budou mít hodnotu 0) a položku `call_id`.

4.1.4 Fáze 3 – korelace získaných dat

Z předchozích dvou fází získáme informace o spojeních mezi veřejnými identitami IMPU a vazbu těchto identit na soukromé identity IMPI. Tyto informace propojíme s ohledem na časovou rovinu, tj. vazba mezi IMPI a IMPU je platná pouze v době od úspěšné registrace do úspěšného odhlášení. Spojení mimo tento interval dokážeme přiřadit pouze veřejné identitě IMPU. Pokud nám bude chybět identita IMPI, přiřadíme ke spojení pouze identitu IMPU. Tímto do záznamů o tocích, jejichž položky jsou uvedeny v tabulce 4.3, doplníme položky `impi1`, `impu1`, `impi2` a `impu2`.

4.1.5 Export získaných dat, rozšíření záznamu IPFIX

Poslední fází je export dat získaných analýzou. Jelikož se nacházíme v prostředí FlowMon, export bude uskutečněn rozšířením formátu IPFIX. Seznam a popis rozšiřujících položek je uveden v tabulce 4.1 spolu s Enterprise ID (identifikátor organizace – použili jsme Enterprise ID VUT) a identifikátorem položky, který musí být unikátní v rámci Enterprise ID. Budeme rozšiřovat záznam toku o čtyři položky – dva páry položek IMPI a IMPU. Jejich význam se bude lišit podle typu toku, který je rozšiřován. Rozlišujeme tyto dva typy toků:

- signalizační tok: přenos signalizačních zpráv mezi UE a funkcí P-CSCF – u tohoto typu toku vyplníme pouze první dvojici položek IMPI a IMPU, a to v případě, že v tomto toku dojde k úspěšné registraci uživatele. Druhá dvojice položek IMPI a IMPU zůstane prázdná.
- datový tok: přenos samotný dat mezi uživateli – u tohoto typu toku vyplníme obě položky IMPU. První položka IMPU bude představovat zdrojového uživatele, druhá pak uživatele cílového. Pokud budou v době začátku přenosu uživatelé zaregistrování, vyplníme i příslušné položky IMPI. Pokud ne, zůstanou tyto položky prázdné.

Záznam o toku již obsahuje informace o množství přenesených dat a době trvání toku, takže je možné provádět i účtování.

Enterprise ID	Item ID	Název položky	Popis
4193	1	IMS_IMPI1	IMPI registrovaného uživatele pro signalizační toky, IMPI zdrojového uživatele pro datové toky
4193	2	IMS_IMPU1	IMPU registrovaného uživatele pro signalizační toky, IMPU zdrojového uživatele pro datové toky
4193	3	IMS_IMPI2	prázdné pro signalizační toky, IMPI cílového uživatele pro datové toky (může být prázdné pro neznámé uživatele)
4193	4	IMS_IMPU2	prázdné pro signalizační toky, IMPU cílového uživatele pro datové toky

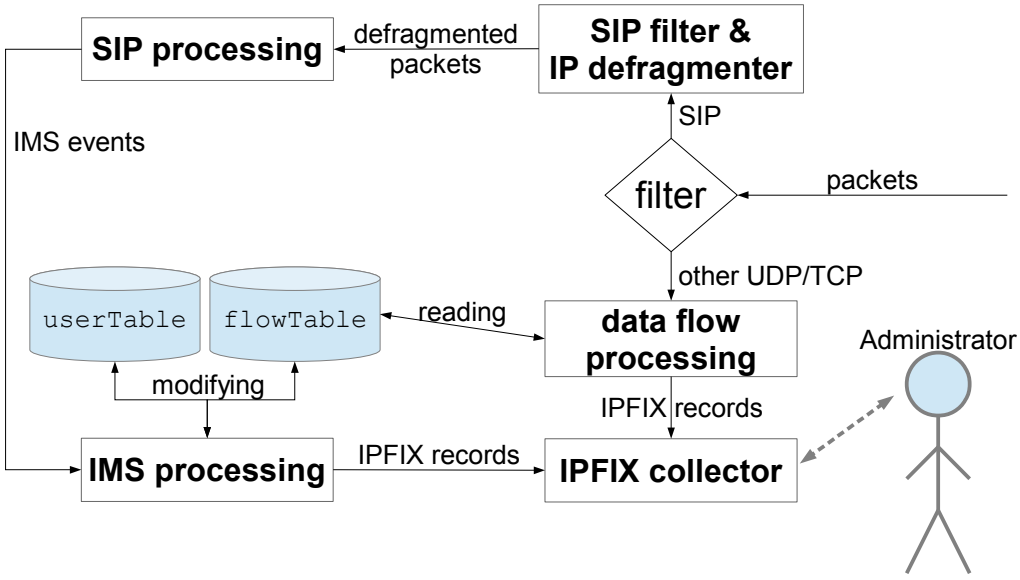
Tabulka 4.1: Rozšiřující položky záznamu toku

4.1.6 Architektura pluginu

Náš nástroj budeme realizovat jako procesní plugin. V procesním pluginu je možné přistupovat k datům celého paketu a paket je zpracován až do úrovně transportní vrstvy. Nemusíme

se tak starat o zpracování těchto vrstev a můžeme se soustředit na vrstvu aplikační. Signalizační zprávy protokolu SIP jsou však často delší než maximální přenosová jednotka na vrstvě IP (MTU) a jsou přenášeny pomocí protokolu UDP, takže může právě na vrstvě IP může docházet k fragmentaci těchto zpráv. FlowMon sonda nepodporuje defragmentaci na vrstvě IP, takže tuto funkcionalitu budeme muset přidat.

Na obrázku 4.2 je zobrazena architekturu pluginu. Přicházející pakety jsou filtrovány a pro pakety protokolu SIP je podle potřeby provedena případná defragmentace. Z paketů jsou získávány podstatné informace (identifikátory, IP adresy, čísla portů, apod.) a tím vznikají události IMS – registrace uživatele, žádost o hovor, ukončení hovoru, odhlášení uživatele, apod. Informacemi z těchto událostí jsou upravovány položky v tabulce uživatelů – `userTable`, která poskytuje vazbu mezi identifikátory IMPI a IMPU, a v tabulce datových toků — `flowTable`, která pro každý tok, který obsahuje, poskytuje kompletní informace o identifikátorech, které jsou dostupné. Pakety, které nejsou identifikovány jako pakety protokolu SIP, jsou podle IP adresy a portu vyhledány v tabulce datových toků. Pokud tabulka datových toků obsahuje vyhledávaný tok, jsou informace z tohoto toku zkopírovány do záznamu toku, do kterého paket přísluší. V tabulce 4.2 uvádíme atributy záznamů registrovaných uživatelů, které se vyskytují v tabulce uživatelů (`userTable`) a v tabulce 4.3 pak uvádíme atributy datových toků z tabulky `flowTable`.



Obrázek 4.2: Architektura nástroje

Atribut	Význam
impi	IMPI - soukromá identita uživatele
impu	IMPU - veřejná identita uživatele

Tabulka 4.2: Atributy záznamů registovaných uživatelů

Atribut	Význam
address	ohlášená IP adresa toku
port	ohlášený port toku RTP – audio
videoPort	ohlášený port toku RTP – video
msrpPort	ohlášený port toku MSRP – přenos souboru
call_id	Call-ID spojení, ke kterému tok přísluší
impi1	IMPI zdrojového uživatele
impu1	IMPU zdrojového uživatele
impi2	IMPI cílového uživatele
impu2	IMPU cílového uživatele

Tabulka 4.3: Atributy záznamů o datových tocích

4.2 Konfigurace prostředí FlowMon sondy

Pro správné zpracování rozšířeného záznamu IPFIX je potřeba provést změny na více místech FlowMon sondy. Tyto změny si nyní přesně popíšeme a ukážeme, co přesně jsme ve kterých souborech přidali či změnili. Všechny konfigurační soubory jsou součástí přiloženého CD spolu s jejich původními verzemi, aby bylo možné zpětně dohledat, jaké změny jsme provedli.

4.2.1 Konfigurace procesu flowmonexp

Základní změnou je přidání nových rozšiřujících položek do konfigurace procesu `flowmonexp`. Tento proces je základním kamenem fungováním FlowMon sondy a dochází v něm k samotnému zpracování síťového provozu. Náš plugin je součástí tohoto procesu.

Do souboru `/etc/flowmon/ipfix-ng_fields.txt` jsme přidali naše rozšiřující položky z tabulky 4.1 tak, že jsme na konec souboru vložili tyto řádky:

```
IMS_IMPI1, 4193, 1, -1
IMS_IMPU1, 4193, 2, -1
```

```
IMS_IMPI2, 4193, 3, -1
IMS_IMPU2, 4193, 4, -1
```

Dále jsme do souboru `/etc/flowmon/ipfix-ng_template.txt` přidali vazbu mezi výše vytvořenými položkami a jejich jménem, které se používá v kódu pluginu. Na konec souboru jsme přidali následující řádky:

```
IMS_IMPI1=IMPI1
IMS_IMPU1=IMPU1
IMS_IMPI2=IMPI2
IMS_IMPU2=IMPU2
```

Tímto je konfigurace procesu `flowmonexp` hotová.

4.2.2 Konfigurace kolektoru ipfixcol

Dále bylo potřeba poskytnout informace o rozšiřujících položkách kolektoru `ipfixcol`[1], aby byl schopný tyto rozšiřující položky správně zpracovat. Položky je nutné identifikovat stejně, jako pro proces flowmonexp, a to pomocí Enterprise ID a Item ID. Je důležité, aby byly tyto identifikátory nakonfigurovány pro oba procesy stejně, aby mohlo dojít k jejich provázání. Položky záznamů IPFIX jsou popsány v souboru `/etc/ipfixcol/ipfix-elements.xml`, do kterého jsme přidali následující řádky:

```
  <element>
    <enterprise>4193</enterprise>
    <id>1</id>
    <name>IMPI1</name>
    <dataType>string</dataType>
    <semantic></semantic>
  </element>
  <element>
    <enterprise>4193</enterprise>
    <id>2</id>
    <name>IMPU1</name>
    <dataType>string</dataType>
    <semantic></semantic>
  </element>
  <element>
    <enterprise>4193</enterprise>
    <id>3</id>
    <name>IMPI2</name>
    <dataType>string</dataType>
    <semantic></semantic>
  </element>
  <element>
    <enterprise>4193</enterprise>
    <id>4</id>
    <name>IMPU2</name>
```

[1] https://www.liberouter.org/technologies/ipfixcol/

```
    <dataType>string</dataType>
    <semantic></semantic>
  </element>
```

Protože jsme chtěli pro zobrazení exportovaných záznamů využít nástroj `fbitdump`[2], bylo ještě nutné přidat do konfigurace kolektoru export záznamů do formátu, který nástroj `fbitdump` podporuje, což je formát `FastBit`[3]. V konfiguraci našeho kolektoru `ipfixcol` však chybělo načítání knihovny `FastBit` a konfigurace exportu do tohoto formátu. Pro načítání knihovny `FastBit` jsme přidali do souboru `/etc/ipfixcol/internalcfg.xml` následující řádky:

```
<storagePlugin>
  <fileFormat>fastbit</fileFormat>
  <file>/usr/share/ipfixcol/plugins/ipfixcol-fastbit-output.so
      </file>
  <threadName>fastbit</threadName>
</storagePlugin>
```

V souboru `/etc/ipfixcol/startup.xml` se nachází konfigurace samotných sběrných a exportních procesů. V původním souboru se nacházely tři sběrné procesy (přijímající data pomocí protokolů UDP, TCP a SCTP). Protože nám stačí pouze sběrný proces pro protokol UDP, ponechali jsme pouze ten s tím, že naslouchá na portu 4000. Zde je kompletní konfigurace sběrného procesu:

```
<collectingProcess>
  <name>UDP collector</name>
  <udpCollector>
    <name>Listening port 4000</name>
    <localPort>4000</localPort>
    <templateLifeTime>1800</templateLifeTime>
    <optionsTemplateLifeTime>1800</optionsTemplateLifeTime>
    <localIPAddress>127.0.0.1</localIPAddress>
  </udpCollector>
  <exportingProcess>FastBit</exportingProcess>
</collectingProcess>
```

V konfiguraci exportního procesu se kromě jiných parametrů nastavuje i formát pojmenovávání výstupních dat. Naše konfigurace vytvoří v místě spuštění procesu složku `output`, ve které se následně vytvoří stromová struktura `rok/měsíc/den` v závislosti na časech exportovaných toků. Kompletní konfigurace exportního procesu:

```
<exportingProcess>
  <name>FastBit</name>
  <destination>
    <name>store data records in FastBit database</name>
    <fileWriter>
```

[2] https://github.com/CESNET/ipfixcol/tree/master/tools/fbitdump
[3] https://github.com/CESNET/libfastbit

```
      <fileFormat>fastbit</fileFormat>
      <path>./output/%Y/%m/%d/</path>
      <dumpInterval>
        <timeWindow>300</timeWindow>
        <timeAlignment>yes</timeAlignment>
        <recordLimit>no</recordLimit>
        <bufferSize>75000</bufferSize>
      </dumpInterval>
      <namingStrategy>
        <type>time</type>
        <prefix>ic</prefix>
      </namingStrategy>
      <reorder>no</reorder>
      <onTheFlyIndexes>no</onTheFlyIndexes>
    </fileWriter>
  </destination>
</exportingProcess>
```

4.2.3 Konfigurace nástroje fbitdump

Poslední částí je konfigurace nástroje `fbitdump`, který využíváme pro zobrazení exportovaných záznamů. I ten totiž potřebuje být nakonfigurován, aby dokázal korektně zobrazit uložená data. Konfigurace se nachází v souboru `/usr/share/fbitdump/fbitdump.xml`, která kromě jiného obsahuje část konfigurující položky (<`columns`> </`columns`>) a část konfigurující výpis informací (<`output`> </`output`>). Do části konfigurující položky jsme přidali tyto řádky:

```
<column>
  <name>IMS IMPI1</name>
  <alias>%impi1</alias>
  <width>18</width>
  <default-value>-</default-value>
  <value type="plain"> <element>e4193id1</element> </value>
</column>
<column>
  <name>IMS IMPU1</name>
  <alias>%impu1</alias>
  <width>18</width>
  <default-value>-</default-value>
  <value type="plain"> <element>e4193id2</element> </value>
</column>
<column>
  <name>IMS IMPI2</name>
  <alias>%impi2</alias>
  <width>18</width>
  <default-value>-</default-value>
  <value type="plain"> <element>e4193id3</element> </value>
</column>
<column>
```

```
  <name>IMS IMPU2</name>
  <alias>%impu2</alias>
  <width>18</width>
  <default-value>-</default-value>
  <value type="plain"> <element>e4193id4</element> </value>
</column>
```

Do části konfigurující výpis informací jsme přidali následující formát výstupu:

Délka toku, počet přenesených bytů, zdrojová IP adresa, zdrojový port, cílová IP adresa, cílový port, IMPI1, IMPU1, IMPI2, IMPU2. Konkrétní přidané řádky jsou tyto:

```
<format>
  <formatName>ims</formatName>
  <formatString>%duration,%byt,%pr,%sa4,%sp,%da4,%dp,%impi1,
      %impu1,%impi2,%impu2</formatString>
</format>
```

4.3 Implementace nástroje

Sonda FlowMon pasivně monitoruje provoz v síti a je většinou zapojena na vstupním či výstupním bodu sítě nebo také na kritických místech, kde chceme monitorovat provoz a zde zpracovává procházející pakety. Plugin pro detekci, monitorování a účtování spojení IMS neslouží k ochraně sítě. Jeho využití spíše přichází v úvahu pro poskytovatele Internetového připojení, pokud by chtěl poskytovatel Internetového připojení sledovat, zda přes jeho síť neprochází provoz jiného poskytovatele bez toho, aniž by si toho byl vědom.

Po získání všech potřebných informací a znalostí můžeme přejít k implementaci, kterou se zabývá tato kapitola spolu s principy fungování pluginu a jeho výstupy. Plugin je implementován v jazyce C s využitím prostředků poskytovaných platformou FlowMon sondy — zpracováváme pouze aplikační data a získané informace vyplňujeme do rozšířených záznamů toků formátu IPFIX.

4.3.1 Popis implementace nástroje

Po načtení pluginu dochází k jeho inicializaci ve funkci `plugin_process_init()`, která alokuje strukturu `ims_process_private_t`, udržující informace potřebné po dobu běhu pluginu. Dále v této funkci dochází k registraci položek IMPI a IMPU rozšiřujících záznamy IPFIX a funkcí, které tyto položky naplňují hodnotami. Druhá inicializační funkce `plugin_process_init_queue()` slouží k inicializaci procesní fronty. Pro nás je zde podstatné, že si můžeme uchovat ID této fronty, neboť jej budeme potřebovat, až budeme vyhledávat reverzní záznamy k tokům a jejich záznamům. Prostředí FlowMon sondy totiž

pracuje s toky odděleně (jeden tok je brán jako dva toky v různých směrech), což nám nevyhovuje, neboť potřebujeme toky provázat z důvodu jejich vzájemné závislosti. Struktury využité v pluginu:

- `ims_rtp_address_t` – struktura uchovávající informace pro zpracování datových toků (IP adresa a port toku, získané identifikátory IMS, Call-ID hovoru, ke kterému tok přísluší).
- `ims_user_t` – struktura uchovávající informace o uživatelích (identifikátory IMPI a IMPU).
- `ims_ip_fragment_t` – struktura pro uchovávání informací a dat nutných pro defragmentaci na vrstvě IP
- `ims_event_t` – struktura události IMS. Udržuje všechny informace potřebné při zpracování událostí v tocích IMS.
- `ims_record_t` – privátní struktura pluginu rozšiřující záznam toku v paměti FlowMon cache. Uchovává poslední zpracovanou událost `ims_event_t` v toku a řetězce pro uložení identit IMPI a IMPU.
- `ims_process_private_t` – privátní struktura pluginu. Uchovává offset privátní struktury záznamu toku `ims_record_t`, ID procesní fronty, tabulku registrovaných uživatelů a datových toků.

Po inicializaci můžeme přejít ke zpracování dat. O to se starají hlavně tyto funkce:

- `ims_data_process()` – zpracovává aplikační data ze kterých se snaží naplnit strukturu `ims_event_t`. Nejprve ověří, že aplikační data obsahují zprávu protokolu SIP tak, že ověří, že první řádek aplikačních dat obsahuje řetězec `SIP`. Ve zprávě typu `REGISTER` následně vyhledává položky `To`, `username` a `expires`, které jsou důležité pro zjištění, zda se jedná o registraci a kdo se registruje. Další důležitá zpracovávaná zpráva je `200 OK`, která registraci potvrzuje. Tak je detekována úspěšná registrace. Ve všech zprávách jsou pak vyhledávány položky `To` a `From` a případné informace o datových tocích v těle zprávy – protokolu SDP.
- `ims_event_process()` – zpracovává událost z předchozí funkce. Pokud se jedná o úspěšnou registraci, zkopíruje řetězce odpovídající identitám IMPI a IMPU z události do privátní struktury záznamu toku a přidá uživatele do tabulky registrovaných uživatelů. Pokud se jedná o úspěšnou de-registraci, uživatel je z tabulky uživatelů odebrán. Pro ostatní typy událostí, kromě událostí typu `BYE` a `CANCEL`, je do tabulky datových toků přidána informace o novém toku, pokud tuto informaci událost obsahuje. Pro události typu `BYE` a `CANCEL` jsou tyto informace odebrány.

- `plugin_process_create()` – volána při přijetí nového paketu, který vytváří nový záznam toku. Pokud je tok směrován z nebo na port příslušící funkci P-CSCF (4060), je využita funkce `ims_data_process()` pro zpracování dat a vytvoření případné události. Vytvořená událost je následně zpracována funkcí `ims_event_process()`. Pro udržování vazby mezi toky je vyhledán reverzní tok k aktuálně zpracovávanému toku. Pokud je nalezen, zkopíruje se před zpracováváním dat poslední událost IMS z reverzního toku do aktuálního. Po zpracování je nová událost zkopírována opět zpět do reverzního toku.
- `plugin_process_update()` – volána při příchodu každého dalšího paketu příslušícího do daného toku. Protože se zpracování prvního a jakéhokoliv dalšího paketu nijak neliší, je tělo této funkce víceméně totožné s předešlou funkcí `plugin_process_create()`.
- `plugin_process_release()` – volána při ukončení zpracování toku před jeho exportem. V této funci se nic neděje, protože nepotřebujeme záznam toku nijak finalizovat.
- `ims_fragment_*()` – funkce pro zpracování fragmentace na vrstvě IP
- `ims_registered_user_*()` – funkce pro práci s tabulkou registrovaných uživatelů
- `ims_ftp_address_*()` – funkce pro práci s tabulkou datových toků
- `value_*()` – funkce pro naplňování rozšířených položek záznamu toků

4.3.2 Výstup pluginu

FlowMon sonda představuje exportér NetFlow/IPFIX záznamů. Monitoruje tedy síťový provoz a vytvořené záznamy následně odesílá na kolektor. Plugin tak má dva výstupy:

1. Při svém běhu vypisuje plugin do konzole informace o tom, jaké události v síťovém provozu detekoval, ukázka na obrázku 4.3, kde můžeme vidět přihlášení uživatele `alice_private@open-ims.test`, následně byly detekovány adresy a porty datových toků v signalizačních zprávách. Na detekovaných adresách poté opravdu došlo k datovému přenosu. Poté byly detekovány další adresy datových toků a došlo k dalšímu přenosu a přihlášený uživatel se odhlásil.

```
INFO: [IMS] 2015-07-21 02:48:25 user alice_private@open-ims.test [alice@open-ims.test] registered
INFO: [IMS] 2015-07-21 02:48:50 new audio RTP address announcement detected: 192.168.1.248:56346
INFO: [IMS] 2015-07-21 02:48:53 new audio RTP address announcement detected: 192.168.1.21:36990
INFO: [IMS] 2015-07-21 02:48:53 Media flow found between users alice_private@open-ims.test[alice@open-ims.test] and [bob@open-ims.test]
INFO: [IMS] 2015-07-21 02:48:53 Media flow found between users alice_private@open-ims.test[alice@open-ims.test] and [bob@open-ims.test]
INFO: [IMS] 2015-07-21 02:50:05 MSRP/RTP address announcement expired: 192.168.1.21:36990
INFO: [IMS] 2015-07-21 02:50:05 MSRP/RTP address announcement expired: 192.168.1.248:56346
INFO: [IMS] 2015-07-21 02:50:21 new audio RTP address announcement detected: 192.168.1.21:18450
INFO: [IMS] 2015-07-21 02:50:24 new audio RTP address announcement detected: 192.168.1.248:21532
INFO: [IMS] 2015-07-21 02:50:24 Media flow found between users [bob@open-ims.test] and alice_private@open-ims.test[alice@open-ims.test]
INFO: [IMS] 2015-07-21 02:50:24 Media flow found between users [bob@open-ims.test] and alice_private@open-ims.test[alice@open-ims.test]
INFO: [IMS] 2015-07-21 02:50:41 MSRP/RTP address announcement expired: 192.168.1.248:21532
INFO: [IMS] 2015-07-21 02:50:41 MSRP/RTP address announcement expired: 192.168.1.21:18450
INFO: [IMS] 2015-07-21 02:50:44 user alice_private@open-ims.test [alice@open-ims.test] de-registered
```

Obrázek 4.3: Výstup pluginu do konzole

2. Pokud běží plugin souběžně s kolektorem `ipfixcol`, kolektor ukládá přijaté záznamy. Tyto záznamy je pak možné zobrazit pomocí nástroje `fbitdump`. Ukázka výstupu na obrázku 4.4, kde můžeme vidět zobrazené informace o exportovaných tocích, které korespondují s informacemi z běhu pluginu na obrázku 4.3, který byl spuštěn nad stejnými daty.

Obrázek 4.4: Výstup pluginu pomocí nástroje fbitdump

4.3.3 Možná rozšíření

Pluginy pro FlowMon sondu jsou primárně konzolové aplikace a grafické uživatelské rozhraní tedy nebylo cílem této práce. Plugin by se však dal o takové grafické rozhraní rozšířit, které by mohlo například zobrazovat datové toky mezi uživateli ve formě síťového diagramu či slučovat toky patřící do stejných hovorů.

Náš nástroj podporuje pouze protokol IP verze 4, protože jsme neměli síťovou infrastrukturu pro vytvoření testovacích dat s protokolem IP verze 6. Podpora zpracování protokolu IP verze 6 by tedy mohla být dalším možným rozšířením tohoto nástroje.

V exportovaných informacích o tocích nijak nerozlišujeme typ dat. Dalším rozšířením by tedy mohlo být přidání položky do záznamu IPFIX, které by blíže určovalo, o jaký typ toku se jedná – video, audio, přenos souboru, apod.

Kapitola 5

Testování nástroje

Pro testování nástroje vytvoříme sadu testovacích souborů, které budou obsahovat všechny zachycené scénáře komunikace, které nám dostupné nástroje dovolí vytvořit. Jako systém IMS jsme použili ústředu Open IMS Core, jejíž konfiguraci jsme popsali v kapitole 3.1. Pro klienty sítě IMS jsme použili aplikaci IMSDroid pro operační systém Android, jejíž konfiguraci jsme popsali v kapitole 3.2. Komunikace bude zachycena ve formátu pcap[1] na různých místech v síti.

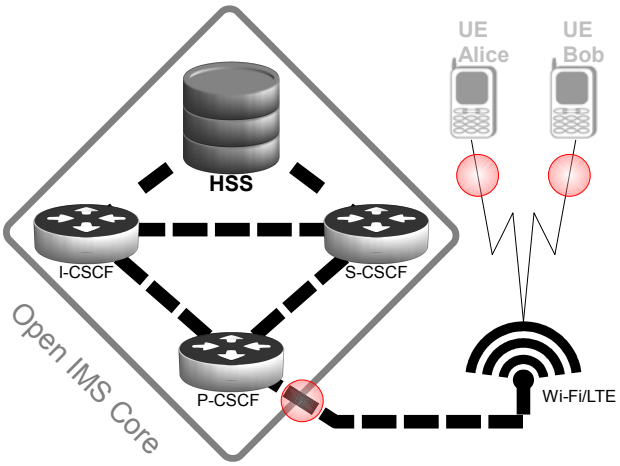
5.1 Testovací síť

Na obrázku 5.1 je zobrazena topologie testovací sítě. UE jsou připojena pomocí bezdrátové sítě Wi-Fi nebo LTE. My jsme měli možnost otestovat pouze Wi-Fi, linková vrstva by však na další vrstvy neměla mít výraznější vliv. K zachytávání souborů pcap dochází jednak na rozhraní systému IMS a také na obou UE. K zachytávání síťového provozu na rozhraní systému IMS byl použit program Wireshark[2], protože systém IMS běží v prostředí virtuálního počítače s operačním systémem Ubuntu. K zachytávání na UE byl použit program Shark for Root[3] pro operační systém Android běžící na každém zařízení zvlášť. Tento program vyžaduje, aby na zařízení, na kterém běží, měl práva uživatele root.

5.2 Testovací data

Vytvořili jsme čtveřici testovacích sad pro čtyři různé scénáře, kde každá sada obsahuje tři soubory zachycené síťové komunikace ve formátu pcap — jeden soubor z rozhraní systému IMS a po jednom souboru z každého UE. Provedli a zachytili jsme tyto scénáře:

[1] http://wiki.wireshark.org/Development/LibpcapFileFormat
[2] https://www.wireshark.org/
[3] https://play.google.com/store/apps/details?id=lv.n3o.shark&hl=en



Obrázek 5.1: Topologie testovací sítě s vyznačenými sběrnými body

- hlasový (audio) hovor
- video-hovor
- výměna textové zprávy
- sdílení souboru

Ve všech čtyřech scénářích jsme nejprve obě UE přihlásili do systému IMS, provedli příslušnou akci (hovor, odeslání zprávy, sdílení souboru) nejprve ve směru z UE Alice na UE Bob, poté stejnou akci ve směru opačném, z UE Bob na UE Alice a nakonec jsme obě UE ze systému IMS odhlásili. Nyní si podrobně popíšeme jednotlivé scénáře a uvedeme, v jakých časových okamžicích a s jakými parametry byly provedené události uskutečněny. Protože zachytáváme síťový provoz na třech nezávislých zařízeních najednou a synchronizaci provádíme ručně, budou se relativní časy událostí od začátku zachytávání lišit. U každého scénáře je proto budeme uvádět zvlášť pro všechny tři vytvořené pcap soubory, a to v těchto sloupcích:

- TS1 – čas události v souboru zachyceném na rozhraní systému IMS
- TS2 – čas události v souboru zachyceném na UE Alice
- TS3 – čas události v souboru zachyceném na UE Bob

Názvy všech vytvořených souborů uvádíme v tabulce 5.1.

5.2.1 Scénář 1 – audio hovor

V tomto scénáři jsme přihlásili obě UE do systému IMS a vytvořili nejprve hovor z UE Alice na UE Bob. Hovor jsme po přibližně 73 s ukončili a vytvořili druhý hovor v opačném

Scénář	Název souboru z rozhraní systému IMS	Název souboru z UE Alice	Název souboru z UE Bob
hovor	audio_core-outside.pcap	audio_alice.pcap	audio_bob.pcap
video-hovor	video_core-outside.pcap	video_alice.pcap	video_bob.pcap
zpráva	message_core-outside.pcap	message_alice.pcap	message_bob.pcap
sdílení	sharing_core-outside.pcap	sharing_alice.pcap	sharing_bob.pcap

Tabulka 5.1: Názvy všech vytvořených testovacích souborů

směru, z UE Alice na UE Bob. Tento hovor jsme po přibližně 17 s ukončili a obě UE jsme ze systému IMS odhlásili. Tabulka 5.2 obsahuje seznam všech podstatných provedených událostí v tomto scénáři spolu s jejich relativními časy.

Událost	TS1	TS2	TS3	Poznámky
přihlášení uživatele Alice	27 s	29 s		
přihlášení uživatele Bob	33 s		33 s	
hovor od Alice k Bobovi	54-127 s	56-129 s	54-127 s	vznikly dva toky protokolu RTP mezi koncovými body 192.168.1.21:36990 a 192.168.1.248:56346
hovor od Boba k Alici	146-163 s	148-165 s	146-163 s	vznikly dva toky protokolu RTP mezi koncovými body 192.168.1.21:18450 and 192.168.1.248:21532
odhlášení uživatele Alice	166 s	168 s		
odhlášení uživatele Bob	172 s		172 s	

Tabulka 5.2: Seznam událostí provedených ve scénáři 1 včetně jejich relativního času

5.2.2 Scénář 2 – video hovor

Tento scénář je podobný jako předchozí scénář, místo běžného hovoru jsme však vytvořili video-hovor. Po přihlášení obou UE do systému IMS jsme vytvořili hovor z UE Alice na UE Bob, který jsme po přibližně 30 s ukončili tím, že jsme odhlásili UE Bob, protože použitý klient IMSDroid nám hovor neumožnil ukončit jinak. Přihlásili jsme tedy UE Bob zpět do systému IMS a provedli druhý hovor z UE Bob na UE Alice, který jsme ukončili přibližně po 25 s, opět odhlášením UE Bob. Nakonec jsme odhlásili UE Alice. Tabulka 5.3 obsahuje seznam všech provedených událostí v tomto scénáři spolu s jejich relativními časy.

Událost	TS1	TS2	TS3	Poznámky
přihlášení uživatele Alice	16 s	19 s		
přihlášení uživatele Bob	20 s		18 s	
hovor od Alice k Bobovi	34-64 s	37-67 s	32-62 s	vznikly dva toky RTP pro audio mezi koncovými body 192.168.1.21:43420 192.168.1.22:24686 a dva toky pro video mezi koncovými body 192.168.1.21:2028 a 192.168.1.22:3774
odhlášení uživatele Bob	65 s		63 s	
přihlášení uživatele Bob	69 s		67 s	
hovor od Boba k Alici	78-103 s	82-106 s	76-101 s	vznikly dva toky RTP pro audio mezi koncovými body 192.168.1.21:8844 a 192.168.1.22:59204 a dva toky pro video mezi koncovými body 192.168.1.21:63580 a 192.168.1.22:46096
odhlášení uživatele Bob	105 s		103 s	
odhlášení uživatele Alice	109 s	112 s		

Tabulka 5.3: Seznam událostí provedených ve scénáři 2 včetně jejich relativních časů

5.2.3 Scénář 3 – textová zpráva

Další funkce, kterou nám klient IMSDroid nabízel, bylo posílání textových zpráv – chat. Stejně, jako v předchozích scénářích, jsme nejprve přihlásili obě UE do systému IMS. Z UE Alice jsme pak na UE Bob poslali zprávu s textem `to Bob` a z UE Bob jsme následně na UE Alice poslali zprávu s textem `to Alice`. Poté jsme obě UE odhlásili ze systému IMS. Tabulka 5.4 obsahuje seznam všech podstatných provedených událostí v tomto scénáři spolu s jejich relativními časy.

Událost	TS1	TS2	TS3	Poznámky
přihlášení uživatele Alice	12 s	10 s		
přihlášení uživatele Bob	14 s		9 s	
zpráva od Alice k Bobovi	23 s	21 s	18 s	text zprávy: `to Bob`
zpráva od Boba k Alici	33 s	31 s	29 s	text zprávy: `to Alice`
odhlášení uživatele Alice	38 s	37 s		
odhlášení uživatele Bob	40 s		35 s	

Tabulka 5.4: Seznam událostí provedených ve scénáři 3 včetně jejich relativního času

5.2.4 Scénář 4 – sdílení souborů

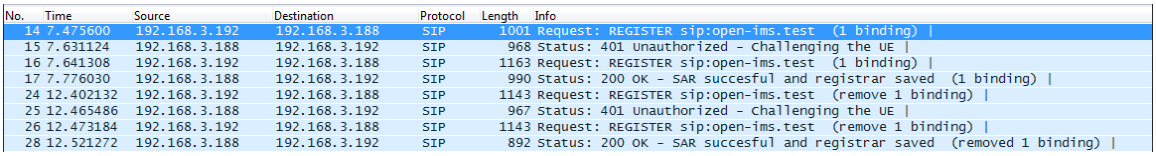
Posledním možným scénářem, který bylo pomocí klienta IMSDroid možné provést, bylo sdílení souborů. Přihlásili jsme tedy nejprve obě UE do systému IMS a následně přenesli z UE Alice na UE Bob soubor `video_alice.pcap` o velikosti 5 300 224 B, přenos trval přibližně 132 s. Následně jsme z UE Bob přenesli na UE Alice soubor `audio_bob.pcap` o velikosti 2 142 893 B, který se přenášel přibližně 54 s. Jak jsme zjistili, k přenosu souborů byl využit protokol Message Session Relay Protocol (MSRP) [2]. Následně jsme obě UE ze systému IMS odhlásili. Tabulka 5.5 obsahuje seznam všech podstatných provedených událostí v tomto scénáři spolu s jejich relativními časy.

Událost	**TS1**	**TS2**	**TS3**	**Poznámky**
přihlášení uživatele Alice	8 s	7 s		
přihlášení uživatele Bob	9 s		9 s	
soubor video_alice.pcap přenesen od Alice k Bobovi	27-159 s	27-159 s	28-159 s	vznikly dva toky protokolu MSRP mezi koncovými body 192.168.1.21:58281 a 192.168.1.248:54900
soubor audio_bob.pcap přenesen od Boba k Alici	194-248 s	194-248 s	194-248 s	vznikly dva toky protokolu MSRP mezi koncovými body 192.168.1.21:39746 a 192.168.1.248:58367
odhlášení uživatele Alice	254 s	254 s		
odhlášení uživatele Bob	256 s		256 s	

Tabulka 5.5: Seznam událostí provedených ve scénáři 4 včetně jejich relativního času

5.2.5 Ukázka dat

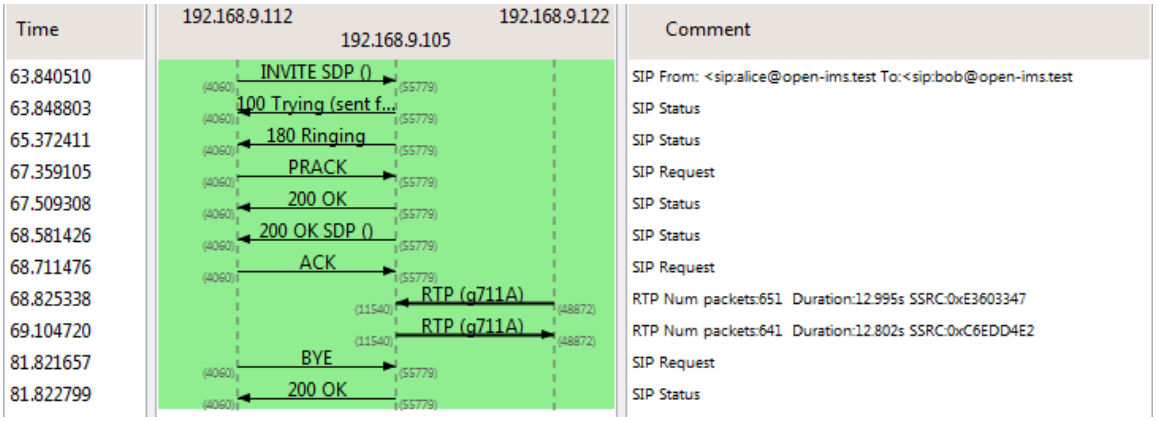
Na obrázcích 5.2, 5.3 a 5.4 jsou zobrazeny ukázky zachycených dat z prvního scénáře. Na obrázku 5.2 jsou zobrazeny zprávy z průběhu přihlášení a odhlášení uživatele. Na obrázku 5.3 jsou zobrazeny zprávy a tok RTP z hovoru mezi UE Alice a UE Bob. Na obrázku 5.4 je tentýž hovor, zprávy jsou však zobrazeny ve formě sekvenčního diagramu.



Obrázek 5.2: Ukázka zachycené síťové komunikace, přihlášení a odhlášení uživatele

No.	Time	Source	Destination	Protocol	Length	Info
355	63.840510	192.168.9.112	192.168.9.105	SIP/SDP	678	Request: INVITE sip:bob@192.168.9.105:55779;transport=udp |
356	63.848803	192.168.9.105	192.168.9.112	SIP	857	Status: 100 Trying (sent from the Transaction Layer) |
361	65.372411	192.168.9.105	192.168.9.112	SIP	1008	Status: 180 Ringing |
430	67.359105	192.168.9.112	192.168.9.105	SIP	1321	Request: PRACK sip:bob@192.168.9.105:55779;transport=udp |
446	67.509308	192.168.9.105	192.168.9.112	SIP	681	Status: 200 OK |
459	68.581426	192.168.9.105	192.168.9.112	SIP/SDP	1447	Status: 200 OK |
460	68.711476	192.168.9.112	192.168.9.105	SIP	1200	Request: ACK sip:bob@192.168.9.105:55779;transport=udp |
467	68.825338	192.168.9.122	192.168.9.105	RTP	216	PT=ITU-T G.711 PCMA, SSRC=0xE3603347, Seq=24178, Time=2094947051
468	68.825565	192.168.9.122	192.168.9.105	RTP	216	PT=ITU-T G.711 PCMA, SSRC=0xE3603347, Seq=24179, Time=2094947211
469	68.839821	192.168.9.122	192.168.9.105	RTP	216	PT=ITU-T G.711 PCMA, SSRC=0xE3603347, Seq=24180, Time=2094947371
470	68.860165	192.168.9.122	192.168.9.105	RTP	216	PT=ITU-T G.711 PCMA, SSRC=0xE3603347, Seq=24181, Time=2094947531
471	68.880506	192.168.9.122	192.168.9.105	RTP	216	PT=ITU-T G.711 PCMA, SSRC=0xE3603347, Seq=24182, Time=2094947691
472	68.900418	192.168.9.122	192.168.9.105	RTP	216	PT=ITU-T G.711 PCMA, SSRC=0xE3603347, Seq=24183, Time=2094947851
473	68.941854	192.168.9.122	192.168.9.105	RTP	216	PT=ITU-T G.711 PCMA, SSRC=0xE3603347, Seq=24184, Time=2094948011
474	68.960840	192.168.9.122	192.168.9.105	RTP	216	PT=ITU-T G.711 PCMA, SSRC=0xE3603347, Seq=24185, Time=2094948171
475	68.982572	192.168.9.122	192.168.9.105	RTP	216	PT=ITU-T G.711 PCMA, SSRC=0xE3603347, Seq=24186, Time=2094948331
476	69.001683	192.168.9.122	192.168.9.105	RTP	216	PT=ITU-T G.711 PCMA, SSRC=0xE3603347, Seq=24187, Time=2094948491

Obrázek 5.3: Ukázka zachycené síťové komunikace, příchozí hovor na UE Bob



Obrázek 5.4: Ukázka zachycené síťové komunikace, příchozí hovor na UE Bob

5.3 Výsledky testů

Každý vytvořený testovací soubor pcap jsme použili jako vstupní data pro námi vytvořený plugin pro FlowMon sondu, který tato data zpracoval. Pomocí kolektoru `ipfixcol` jsme pak zpracovaná data exportovali a pomocí nástroje `fbitdump` zobrazili v textové podobě. Všechny výsledky jsou seřazeny vzestupně podle času začátku toku pomocí přepínače `-m` nástroje `fbitdump`. Nyní si výsledky těchto testů ukážeme a popíšeme.

Signalizační zprávy jsou poměrně dlouhé, často delší než maximální přenosová jednotka (MTU) na vrstvě IP, a jsou přenášeny pomocí protokolu UDP. Proto dochází k jejich fragmentaci už na vrstvě IP, takže se v rámci jednoho toku vyskytuje více různých hlaviček IP – s různou hodnotou položky `Identification`. FlowMon sonda bohužel používá i tuto položku k rozlišení toků, takže se nám jeden tok rozpadne na víc toků, i když mají stejné cílové i zdrojové adresy a porty. Máme dvě možnosti, jak tento problém řešit. První variantou je zkopírovat získané informace do všech toků, do kterých se daný tok rozpadl. Tím však dojde k duplikaci stejné informace do více toků. Druhá možnost je ponechat získané informace pouze v jednom toku s tím, že ztratíme informace o tom, kolik signalizačních dat

bylo přeneseno a jak dlouho celý tok trval (jak dlouho byl uživatel přihlášen). Spokojíme se s první možností, protože signalizační toky nejsou naším hlavním cílem, důležité jsou z našeho pohledu toky, ve kterých dochází k přenosu samotných dat. Výpisy nástroje `fbitdump` tak budou přehlednější, i když informace u signalizačních toků nemusí být přesné. Pokud se signalizační tok rozpadne na víc toků, časová délka toku bude kratší a počet přenesených bytů menší, než tomu bylo ve skutečnosti.

5.3.1 Scénář 1 – audio hovor

Testování tohoto scénáře jsme provedli na všech třech vstupních souborech odděleně. Výsledný vyfiltrovaný výstup můžeme vidět na obrázcích 5.5, 5.6 a 5.7 – vyfiltrovali jsme pouze ty toky, u kterých došlo k detekci uživatelů. Obrázek 5.5 zobrazuje výpis po zpracování souboru `audio_core-outside.pcap`, který byl zachycen na rozhraní systému IMS. Na tomto výpisu vidíme záznamy o čtyřech tocích spolu s jejich délkou trváním a počtu bytů, které byly těmito toky přeneseny. Vstupní soubor neobsahoval samotné datové toky, protože ty neprochází systémem IMS, ale přímo mezi uživateli. Můžeme tedy pozorovat pouze signalizační toky obou uživatelů, které nejsou kompletní z důvodu, který jsme si popsali výše. Vidíme však, že se nám podařilo úspěšně detekovat přihlášení obou uživatelů.

```
Duration, Bytes,Proto,     Src IPv4,sPort,     Dst IPv4,dPort,                IMS IMPI1,          IMS IMPU1,                IMS IMPI2,          IMS IMPU2
   0.083,  2139,  UDP,192.168.1.248,39070, 192.168.1.20,4060 ,alice_private@open-ims.test,alice@open-ims.test,                        -,                  -
  47.424,  5981,  UDP, 192.168.1.20,4060 ,192.168.1.248,39070,alice_private@open-ims.test,alice@open-ims.test,                        -,                  -
   0.960,  4294,  UDP, 192.168.1.21,53931, 192.168.1.20,4060 ,  bob_private@open-ims.test,  bob@open-ims.test,                        -,                  -
  31.945,  9776,  UDP, 192.168.1.20,4060 , 192.168.1.21,53931,  bob_private@open-ims.test,  bob@open-ims.test,                        -,                  -
```

Obrázek 5.5: Výpis nástroje fbitdump po zpracování souboru audio_core-outside.pcap

Další testovaný soubor `audio_alice.pcap`, který byl zachycen na UE Alice, obsahuje jak signalizační tok pro uživatele `alice@open-ims.test`, tak oba multimediální toky protokolu RTP příslušící ke dvěma provedeným hovorům. Tyto toky už se nerozpadají, a tak jsou informace kompletní. Výstup z nástroje `fbitdump` můžeme vidět na obrázku 5.6, který zobrazuje šest toků opět s jejich příslušnými délkami a počtem přenesených bytů. První dva toky jsou signalizační, druhé dva jsou datové toky prvního hovoru a druhé dva datové toky druhého hovoru. Délka toků odpovídá délce hovorů a počet přenesených bytů také, což jsme ověřili pomocí programu Wireshark a také tím, že se pro daný vstupní pcap soubor nevyskytují na výstupu duplicitní toky.

```
Duration, Bytes,Proto,     Src IPv4,sPort,     Dst IPv4,dPort,                IMS IMPI1,          IMS IMPU1,                IMS IMPI2,          IMS IMPU2
   0.282,  2139,  UDP,192.168.1.248,39070, 192.168.1.20,4060 ,alice_private@open-ims.test,alice@open-ims.test,                        -,                  -
   7.428,  1965,  UDP, 192.168.1.20,4060 ,192.168.1.248,39070,alice_private@open-ims.test,alice@open-ims.test,                        -,                  -
 72.471,730824,  UDP,192.168.1.248,56346, 192.168.1.21,36990,alice_private@open-ims.test,alice@open-ims.test,                        -,  bob@open-ims.test
 72.121,722056,  UDP, 192.168.1.21,36990,192.168.1.248,56346,alice_private@open-ims.test,alice@open-ims.test,                        -,  bob@open-ims.test
 17.199,167432,  UDP, 192.168.1.21,18450,192.168.1.248,21532,                          -,  bob@open-ims.test,alice_private@open-ims.test,alice@open-ims.test
 16.648,165832,  UDP,192.168.1.248,21532, 192.168.1.21,18450,                          -,  bob@open-ims.test,alice_private@open-ims.test,alice@open-ims.test
```

Obrázek 5.6: Výpis nástroje fbitdump po zpracování souboru audio_alice.pcap

Posledním testovacím souborem v tomto scénáři byl soubor `audio_bob.pcap`, který byl zachycen na UE Bob. Výsledek testu můžem vidět na obrázku 5.7 a je obdobný jako u předchozího souboru s tím rozdílem, že nyní máme k dispozici informace ze signalizačního toku uživatele `bob@open-ims.test`.

```
Duration, Bytes,Proto,     Src IPv4,sPort,     Dst IPv4,dPort,                IMS IMPI1,         IMS IMPU1,                IMS IMPI2,         IMS IMPU2
   1.053,  4294,  UDP, 192.168.1.21,53931, 192.168.1.20,4060 ,  bob_private@open-ims.test,  bob@open-ims.test,                        -,                 -
   1.802,  3878,  UDP, 192.168.1.20,4060 , 192.168.1.21,53931,  bob_private@open-ims.test,  bob@open-ims.test,                        -,                 -
  72.286,730024,  UDP,192.168.1.248,56346, 192.168.1.21,36990,                          -,alice@open-ims.test,  bob_private@open-ims.test,  bob@open-ims.test
  72.121,722256,  UDP, 192.168.1.21,36990,192.168.1.248,56346,                          -,alice@open-ims.test,  bob_private@open-ims.test,  bob@open-ims.test
  17.196,167432,  UDP, 192.168.1.21,18450,192.168.1.248,21532,  bob_private@open-ims.test,  bob@open-ims.test,                        -,alice@open-ims.test
  16.570,165832,  UDP,192.168.1.248,21532, 192.168.1.21,18450,  bob_private@open-ims.test,  bob@open-ims.test,                        -,alice@open-ims.test
```

Obrázek 5.7: Výpis nástroje fbitdump po zpracování souboru audio_bob.pcap

5.3.2 Scénář 2 – video hovor

Tento scénář je obdobný jako scénář předchozí, který zahrnoval audio hovor. V případě video-hovoru však vznikají pro každý hovor toky čtyři – dva pro audio a dva pro video. Vyfiltrovaný výstup pro všechny tři vstupní soubory můžeme vidět na obrázcích 5.8, 5.9 a 5.10. Výstup po zpracování souboru `video_core-outside.pcap`, který byl zachycen na rozhraní systému IMS můžeme vidět na obrázku 5.8, kde vidíme signalizační toky obou uživatelů. U uživatele `bob@open-ims.test` vidíme toky dva, protože se odhlásil a znovu přihlásil, aby byl video-hovor ukončen. Poslední řádek obsahující identity uživatele `alice@open-ims.test` je další ukázkou rozdělení záznamu toku FlowMon sondou.

```
Duration, Bytes,Proto,     Src IPv4,sPort,     Dst IPv4,dPort,                IMS IMPI1,         IMS IMPU1,                IMS IMPI2,         IMS IMPU2
   0.060,  2141,  UDP, 192.168.1.22,37658, 192.168.1.20,4060 ,alice_private@open-ims.test,alice@open-ims.test,                        -,                 -
 103.835, 18435,  UDP, 192.168.1.20,4060 , 192.168.1.22,37658,alice_private@open-ims.test,alice@open-ims.test,                        -,                 -
   0.940,  3301,  UDP, 192.168.1.21,55638, 192.168.1.20,4060 ,  bob_private@open-ims.test,  bob@open-ims.test,                        -,                 -
  50.053, 12333,  UDP, 192.168.1.20,4060 , 192.168.1.21,55638,  bob_private@open-ims.test,  bob@open-ims.test,                        -,                 -
  10.602,  7519,  UDP, 192.168.1.21,35481, 192.168.1.20,4060 ,  bob_private@open-ims.test,  bob@open-ims.test,                        -,                 -
  56.956, 14564,  UDP, 192.168.1.20,4060 , 192.168.1.21,35481,  bob_private@open-ims.test,  bob@open-ims.test,                        -,                 -
   6.338,  2924,  UDP, 192.168.1.22,37658, 192.168.1.20,4060 ,alice_private@open-ims.test,alice@open-ims.test,                        -,                 -
```

Obrázek 5.8: Výpis nástroje fbitdump po zpracování souboru video_core-outside.pcap

Na obrázku 5.9 můžeme vidět výsledek testu se vstupním souborem `video_alice.pcap`, který byl zachycen na UE Alice. Vidíme, že se podařilo úspěšně detekovat všechny čtyři datové toky pro každý hovor s délkami odpovídajícími délkám příslušných hovorů.

```
Duration, Bytes,Proto,     Src IPv4,sPort,     Dst IPv4,dPort,                IMS IMPI1,         IMS IMPU1,                IMS IMPI2,         IMS IMPU2
   0.163,  2141,  UDP, 192.168.1.22,37658, 192.168.1.20,4060 ,alice_private@open-ims.test,alice@open-ims.test,                        -,                 -
   3.860,  2037,  UDP, 192.168.1.20,4060 , 192.168.1.22,37658,alice_private@open-ims.test,alice@open-ims.test,                        -,                 -
  30.273,302744,  UDP, 192.168.1.22,24686, 192.168.1.21,43420,alice_private@open-ims.test,alice@open-ims.test,                        -,  bob@open-ims.test
  30.488, 1.2 M,  UDP, 192.168.1.22,3774 , 192.168.1.21,2028 ,alice_private@open-ims.test,alice@open-ims.test,                        -,  bob@open-ims.test
  30.106,292048,  UDP, 192.168.1.21,43420, 192.168.1.22,24686,alice_private@open-ims.test,alice@open-ims.test,                        -,  bob@open-ims.test
  30.108,928242,  UDP, 192.168.1.21,2028 , 192.168.1.22,3774 ,alice_private@open-ims.test,alice@open-ims.test,                        -,  bob@open-ims.test
  24.452,238888,  UDP, 192.168.1.21,8844 , 192.168.1.22,59204,                          -,  bob@open-ims.test,alice_private@open-ims.test,alice@open-ims.test
  24.439,735814,  UDP, 192.168.1.21,63580, 192.168.1.22,46096,                          -,  bob@open-ims.test,alice_private@open-ims.test,alice@open-ims.test
  24.029,239760,  UDP, 192.168.1.22,59204, 192.168.1.21,8844 ,                          -,  bob@open-ims.test,alice_private@open-ims.test,alice@open-ims.test
  24.237,900644,  UDP, 192.168.1.22,46096, 192.168.1.21,63580,                          -,  bob@open-ims.test,alice_private@open-ims.test,alice@open-ims.test
  10.118,  3293,  UDP, 192.168.1.20,4060 , 192.168.1.22,37658,alice_private@open-ims.test,alice@open-ims.test,                        -,                 -
   6.341,  2924,  UDP, 192.168.1.22,37658, 192.168.1.20,4060 ,alice_private@open-ims.test,alice@open-ims.test,                        -,                 -
```

Obrázek 5.9: Výpis nástroje fbitdump po zpracování souboru video_alice.pcap

Na obrázku 5.10 pak vidíme výsledek testu posledního souboru v tomto scénáři -

`video_bob.pcap`. Výsledek je podobný předchozímu souboru, opět s tím rozdílem, že máme k dispozici informace ze signalizačního toku uživatele `bob@open-ims.test`.

```
Duration, Bytes,Proto,     Src IPv4,sPort,     Dst IPv4,dPort,                IMS IMPI1,          IMS IMPU1,                IMS IMPI2,          IMS IMPU2
   0.957,  3301,  UDP, 192.168.1.21,55638, 192.168.1.20,4060 ,  bob_private@open-ims.test,  bob@open-ims.test,                        -,                  -
  19.761,  9486,  UDP, 192.168.1.20,4060 , 192.168.1.21,55638,  bob_private@open-ims.test,  bob@open-ims.test,                        -,                  -
  30.239, 1.2 M,  UDP, 192.168.1.22,3774 , 192.168.1.21,2028 ,                           -,alice@open-ims.test,  bob_private@open-ims.test,  bob@open-ims.test
  30.010,302344,  UDP, 192.168.1.22,24686, 192.168.1.21,43420,                           -,alice@open-ims.test,  bob_private@open-ims.test,  bob@open-ims.test
  30.108,293448,  UDP, 192.168.1.21,43420, 192.168.1.22,24686,                           -,alice@open-ims.test,  bob_private@open-ims.test,  bob@open-ims.test
  30.107,933782,  UDP, 192.168.1.21,2028 , 192.168.1.22,3774 ,                           -,alice@open-ims.test,  bob_private@open-ims.test,  bob@open-ims.test
  10.620,  7519,  UDP, 192.168.1.21,35481, 192.168.1.20,4060 ,  bob_private@open-ims.test,  bob@open-ims.test,                        -,                  -
  11.143,  7230,  UDP, 192.168.1.20,4060 , 192.168.1.21,35481,  bob_private@open-ims.test,  bob@open-ims.test,                        -,                  -
  24.453,240288,  UDP, 192.168.1.21,8844 , 192.168.1.22,59204,  bob_private@open-ims.test,  bob@open-ims.test,                        -,alice@open-ims.test
  24.440,748967,  UDP, 192.168.1.21,63580, 192.168.1.22,46096,  bob_private@open-ims.test,  bob@open-ims.test,                        -,alice@open-ims.test
  24.031,239760,  UDP, 192.168.1.22,59204, 192.168.1.21,8844 ,  bob_private@open-ims.test,  bob@open-ims.test,                        -,alice@open-ims.test
  24.246,899854,  UDP, 192.168.1.22,46096, 192.168.1.21,63580,  bob_private@open-ims.test,  bob@open-ims.test,                        -,alice@open-ims.test
```

Obrázek 5.10: Výpis nástroje fbitdump po zpracování souboru video_bob.pcap

5.3.3 Scénář 3 – textová zpráva

V tomto scénáři jsme posílali textové zprávy mezi uživateli. Textové zprávy se přenáší v rámci signalizačních toků, takže se nijak neprojeví na výstupu – můžeme pozorovat pouze signalizační toky tak jako ve všech ostatních scénářích. Na obrázcích 5.11, 5.12 a 5.13 můžeme vidět výsledky pro všechny tři vstupní soubory – `message_core-outside.pcap`, `message_alice.pcap` a `message_bob.pcap`.

```
Duration, Bytes,Proto,     Src IPv4,sPort,     Dst IPv4,dPort,                IMS IMPI1,          IMS IMPU1,                IMS IMPI2,          IMS IMPU2
   0.071,  2139,  UDP, 192.168.1.22,38726, 192.168.1.20,4060 ,alice_private@open-ims.test,alice@open-ims.test,                        -,                  -
  34.263,  5476,  UDP, 192.168.1.20,4060 , 192.168.1.22,38726,alice_private@open-ims.test,alice@open-ims.test,                        -,                  -
   9.298,  2743,  UDP, 192.168.1.21,50632, 192.168.1.20,4060 ,  bob_private@open-ims.test,  bob@open-ims.test,                        -,                  -
  32.691,  5443,  UDP, 192.168.1.20,4060 , 192.168.1.21,50632,  bob_private@open-ims.test,  bob@open-ims.test,                        -,                  -
```

Obrázek 5.11: Výpis nástroje fbitdump po zpracování souboru message_core-outside.pcap

```
Duration, Bytes,Proto,     Src IPv4,sPort,     Dst IPv4,dPort,                IMS IMPI1,          IMS IMPU1,                IMS IMPI2,          IMS IMPU2
   0.206,  2139,  UDP, 192.168.1.22,38726, 192.168.1.20,4060 ,alice_private@open-ims.test,alice@open-ims.test,                        -,                  -
  26.475,  5446,  UDP, 192.168.1.20,4060 , 192.168.1.22,38726,alice_private@open-ims.test,alice@open-ims.test,                        -,                  -
```

Obrázek 5.12: Výpis nástroje fbitdump po zpracování souboru message_alice.pcap

```
Duration, Bytes,Proto,     Src IPv4,sPort,     Dst IPv4,dPort,                IMS IMPI1,          IMS IMPU1,                IMS IMPI2,          IMS IMPU2
   9.406,  2743,  UDP, 192.168.1.21,50632, 192.168.1.20,4060 ,  bob_private@open-ims.test,  bob@open-ims.test,                        -,                  -
  32.712,  5443,  UDP, 192.168.1.20,4060 , 192.168.1.21,50632,  bob_private@open-ims.test,  bob@open-ims.test,                        -,                  -
```

Obrázek 5.13: Výpis nástroje fbitdump po zpracování souboru message_bob.pcap

5.3.4 Scénář 4 – sdílení souborů

V posledním scénáři jsme přenášeli soubor od jednoho uživatele k druhému a poté jiný soubor v obráceném směru. Výsledky testu pro všechny vstupní soubory můžeme vidět na obrázcích 5.14, 5.15 a 5.16. Výstup zpracování souboru `sharing_core-outside.pcap` je zobrazen na obrázku 5.14, kde můžeme opět vidět pouze signalizační toky.

```
Duration, Bytes,Proto,     Src IPv4,sPort,     Dst IPv4,dPort,                IMS IMPI1,          IMS IMPU1,                IMS IMPI2,          IMS IMPU2
   0.103,  2140,  UDP, 192.168.1.22,51816, 192.168.1.20,4060 ,alice_private@open-ims.test,alice@open-ims.test,                        -,                  -
  25.816,  4645,  UDP, 192.168.1.20,4060 , 192.168.1.22,51816,alice_private@open-ims.test,alice@open-ims.test,                        -,                  -
   0.366,  2147,  UDP, 192.168.1.21,46093, 192.168.1.20,4060 ,  bob_private@open-ims.test,  bob@open-ims.test,                        -,                  -
  24.150,  6130,  UDP, 192.168.1.20,4060 , 192.168.1.21,46093,  bob_private@open-ims.test,  bob@open-ims.test,                        -,                  -
```

Obrázek 5.14: Výpis nástroje fbitdump po zpracování souboru sharing_core-outside.pcap

Na obrázku 5.15 můžeme vidět výstup zpracování souboru `sharing_alice.pcap`, který byl zachycen na UE Alice. Zde už můžeme vidět i toky (třetí až poslední), ve kterých došlo k přenosu souborů. Množství přenesených paketů odpovídá přibližně 130% velikosti souborů, což zahrnuje i režii použitých protokolů. Na obrázku 5.16 můžeme vidět totéž, ze souboru `sharing_bob.pcap` zachyceného na UE Bob.

```
Duration, Bytes,Proto,     Src IPv4,sPort,     Dst IPv4,dPort,                IMS IMPI1,          IMS IMPU1,                IMS IMPI2,          IMS IMPU2
   0.117,  2140,  UDP, 192.168.1.22,51816, 192.168.1.20,4060 ,alice_private@open-ims.test,alice@open-ims.test,                        -,                  -
  25.830,  4645,  UDP, 192.168.1.20,4060 , 192.168.1.22,51816,alice_private@open-ims.test,alice@open-ims.test,                        -,                  -
 131.499, 6.5 M,  TCP, 192.168.1.22,54900, 192.168.1.21,58281,alice_private@open-ims.test,alice@open-ims.test,                        -,  bob@open-ims.test
 131.505,740024,  TCP, 192.168.1.21,58281, 192.168.1.22,54900,alice_private@open-ims.test,alice@open-ims.test,                        -,  bob@open-ims.test
  54.150,295625,  TCP, 192.168.1.22,58367, 192.168.1.21,39746,                          -,  bob@open-ims.test,alice_private@open-ims.test,alice@open-ims.test
  54.151, 2.6 M,  TCP, 192.168.1.21,39746, 192.168.1.22,58367,                          -,  bob@open-ims.test,alice_private@open-ims.test,alice@open-ims.test
```

Obrázek 5.15: Výpis nástroje fbitdump po zpracování souboru sharing_alice.pcap

```
Duration, Bytes,Proto,     Src IPv4,sPort,     Dst IPv4,dPort,                IMS IMPI1,          IMS IMPU1,                IMS IMPI2,          IMS IMPU2
   0.191,  2147,  UDP, 192.168.1.21,46093, 192.168.1.20,4060 ,  bob_private@open-ims.test,  bob@open-ims.test,                        -,                  -
  24.077,  6130,  UDP, 192.168.1.20,4060 , 192.168.1.21,46093,  bob_private@open-ims.test,  bob@open-ims.test,                        -,                  -
 131.504,741338,  TCP, 192.168.1.21,58281, 192.168.1.22,54900,                          -,alice@open-ims.test,  bob_private@open-ims.test,  bob@open-ims.test
 131.451, 6.5 M,  TCP, 192.168.1.22,54900, 192.168.1.21,58281,                          -,alice@open-ims.test,  bob_private@open-ims.test,  bob@open-ims.test
  54.151,295625,  TCP, 192.168.1.22,58367, 192.168.1.21,39746,  bob_private@open-ims.test,  bob@open-ims.test,                        -,alice@open-ims.test
  54.151, 2.6 M,  TCP, 192.168.1.21,39746, 192.168.1.22,58367,  bob_private@open-ims.test,  bob@open-ims.test,                        -,alice@open-ims.test
```

Obrázek 5.16: Výpis nástroje fbitdump po zpracování souboru sharing_bob.pcap

5.4 Zhodnocení testů

V této kapitole jsme popsali testování nástroje – jaké testy byly prováděny a co bylo jejich výsledkem. Otestovali jsme, že dokážeme správně detekovat všechna datová spojení (toky) a přiřadit k nim správné identity uživatelů, mezi kterými ke spojením docházelo. Při analýze síťového provozu na rozhraní systému IMS vždy dochází ke správné detekci uživatelských identit, ale není možné ve stejném místě detekovat a monitorovat datová spojení, protože tímto místem neprocházejí. Pokud analyzujeme síťový provoz na koncovém bodu UE, dokážeme zjistit pouze identitu IMPI daného koncového bodu, nikoli však koncového bodu, který je na druhé straně toku.

Pro kompletní analýzu potřebujeme buď analyzovat síťový provoz v místě sítě, kterým prochází veškerý provoz mezi všemi zařízeními (což nemusí být možné, protože takové místo v síti nemusí existovat) nebo analýzu provádět na více místech a chybějící položky pro stejné toky vzájemně doplňovat. Takové řešení nám umožňuje analyzovat síťový provoz v celé síti bez ohledu na to, jak je rozsáhlá.

Kapitola 6

Závěr a zhodnocení

Cílem této práce bylo seznámit se s architekturou sítí IMS, prostudovat volně dostupné implementace implementace systému IMS a klientů systému IMS a možnosti jejich využití. Následně bylo cílem vytvořit testovací síť IMS s několika službami a uživateli, navrhnout, implementovat a otestovat nástroj pro monitorování a účtování spojení v sítích IMS.

V teoretické části jsme si popsali části systému IMS, které jsou podstatné pro detekci a monitorování komunikace s tímto systémem. Dále jsme si popsali vytvoření a konfiguraci systému IMS s využitím ústředny Open IMS Core včetně konfigurace klientských zařízení, k čemuž jsme využili aplikaci IMSDroid pro operační systém Android.

Největší pozornost jsme věnovali návrhu a implementaci nástroje pro monitorování a účtování spojení v sítích IMS, kde jsme popsali jeho jednotlivé komponenty a jejich fungování. Popsali jsme úpravy rozšiřující funkcionalitu FlowMon sondy, pro kterou byl nástroj implementován jako plugin. Také jsme uvedli, jaké informace jsou důležité pro správnou analýzu komunikace IMS a jakým způsobem je lze vyhledat v síťovém provozu.

Navržený nástroj jsme implementovali v jazyce C v prostředí FlowMon sondy s využitím zpracování síťových dat poskytovaných tímto prostředím. Podstatnou část práce jsme věnovali testování vytvořeného nástroje. Provedli jsme řadu scénářů představujících různé prováděné činnosti (audio či video-hovor, posílání textových zpráv a sdílení souborů). Síťovou komunikaci mezi zařízeními během těchto scénářů jsme zachytili ve formě souborů ve formátu pcap, které jsme použili pro samotné testování nástroje.

Z hlediska dalšího vývoje by bylo vhodné nástroj rozšířit o vizualizační část a další položky záznamu IPFIX, které by například identifikovaly, o jaký typ spojení se jedná (audio, video, přenos souboru). Rovněž by bylo možné přidat podporu protokolu IPv6, kterou jsme neimplementovali z důvodu nemožnosti vytvořit testovací data.

Literatura

[1] 3GPP: IP Multimedia Subsystem (IMS); Stage 2. TS 23.228, 3rd Generation Partnership Project (3GPP), Září 2008.
URL http://www.3gpp.org/ftp/Specs/html-info/23228.htm

[2] Campbell, B.; Mahy, R.; Jennings, C.: The Message Session Relay Protocol (MSRP). RFC 4975 (Proposed Standard), Září 2007.
URL http://www.ietf.org/rfc/rfc4975.txt

[3] Claise, B.; Trammell, B.; Aitken, P.: Specification of the IP Flow Information Export (IPFIX) Protocol for the Exchange of Flow Information. RFC 7011 (INTERNET STANDARD), Září 2013.
URL http://www.ietf.org/rfc/rfc7011.txt

[4] Handley, M.; Jacobson, V.; Perkins, C.: SDP: Session Description Protocol. RFC 4566 (Proposed Standard), Červenec 2006.
URL http://www.ietf.org/rfc/rfc4566.txt

[5] INVEA-TECH: FlowMon. [online].
URL http://www.invea.cz/products/flowmon

[6] INVEA-TECH: FlowMon. [online].
URL https://www.invea.com/cs/produkty-sluzby/flowmon/flowmon-sondy

[7] Rosenberg, J.; Schulzrinne, H.; Camarillo, G.; aj.: SIP: Session Initiation Protocol. RFC 3261 (Proposed Standard), Červen 2002.
URL http://www.ietf.org/rfc/rfc3261.txt

[8] Schulzrinne, H.; Casner, S.; Frederick, R.; aj.: RTP: A Transport Protocol for Real-Time Applications. RFC 3550 (INTERNET STANDARD), Červenec 2003.
URL http://www.ietf.org/rfc/rfc3550.txt

Příloha A

Obsah CD

Obsahuje zdrojové texty, konfigurační soubory pro FlowMon sondu, manuál, testovací data z laboratoře a výsledky těchto testů.

`plugin/`	zdrojové soubory pluginu
`cfg/`	konfigurační soubory
`tests/`	testovací data
`  case_study_1-audio/`	
`  case_study_2-video/`	
`  case_study_3-message/`	
`  case_study_4-sharing/`	
`tests_results/`	výsledky testů
`  case_study_1-audio/`	
`  case_study_2-video/`	
`  case_study_3-message/`	
`  case_study_4-sharing/`	
`manual.pdf`	manuál k ovládání nástroje

Příloha B

Manuál

Pro spuštění pluginu musíme mít FlowMon sondu verze 7.00 nebo vyšší. Nejprve musíme zdrojové soubory pluginu přeložit, to provedeme příkazem `make` na úrovni složky `plugin`. Příkaz pro spuštění pluginu se skládá z mnoha parametrů. Základním příkazem, který spouští zpracování je příkaz `sudo flowmonexp`, jehož parametry se dají zobrazit přepínačem `-h` a popisy všech jeho pluginů včetně parametrů pak přepínačem `-l`. Parametry, které jsou dostačující pro spuštění našeho pluginu jsou následující:

- `-X` – specifikuje cestu k externímu pluginu – soubor `ims.so`, který vznikl přeložením zdrojových kódů pluginu
- `-I` – určuje vstupní plugin a jeho parametry – pro vstup ze souboru pcap můžeme použít vstupní plugin `pcap-replay`, pro vstup ze síťového rozhraní můžeme použít plugin `rawnetcap`.
- `-P` – určuje procesní plugin, což je náš plugin – `ims`
- `-E` – určuje exportní plugin, použijeme `ipfix-ng`

Ukázka kompletního příkazu pro spuštění pluginu se vstupem ze souboru `soubor.pcap` a exportem směrovaným na localhost, port 4000:

```
$ sudo flowmonexp -X ./ims.so -I pcap-replay:file="soubor.pcap"
,speed=0,loop=1 -P ims -E ipfix-ng:host=localhost,port=4000,
source_id=3,template-refresh-packets=4096,
template-refresh-time=600
```

Ukázka kompletního příkazu pro spuštění pluginu se vstupem ze síťového rozhraní `eth0` a exportem směrovaným na localhost, port4000:

```
$ sudo flowmonexp -X ./ims.so -I rawnetcap:device=eth0,
sampling=0,cache-size=32768 -P ims -E ipfix-ng:host=localhost,
port=4000,source_id=3,template-refresh-packets=4096,
template-refresh-time=600
```

V tomto stavu nám plugin vypisuje na konzolový výstup informace o tom, jaké události detekoval (registrace uživatele, oznámení datových toků, výskyt datových toků, apod.). K tomu, abychom mohli detekované informace exportovat, je potřeba rozšířit záznam formátu IPFIX a mít při běhu pluginu spuštěný kolektor `ipfixcol`. Rozšíření záznamu provedeme zkopírováním konfiguračních souborů z adresáře `cfg` do příslušných umístění. Kolektor `ipfixcol` poté bude exportovat toky do adresáře `output`, odkud je můžeme zobrazit nástrojem `fbitdump` pomocí příkazu

```
$ fbitdump -R output -o ims -m
```